# AR1335

# AR1335 1/3.2-Inch 13 Mp CMOS Digital Image Sensor

**ON Semiconductor®**

www.onsemi.com

This document, and the information contained herein, is CONFIDENTIAL AND PROPRIETARY and the property of Semiconductor Components Industries, LLC., dba ON Semiconductor. It shall not be used, published, disclosed or disseminated outside of the Company, in whole or in part, without the written permission of ON Semiconductor. Reverse engineering of any or all of the information contained herein is strictly prohibited.

© 2016, SCILLC. All Rights Reserved.

## General Description

The AR1335 from ON Semiconductor is a 1/3.2-inch BSI (Back Side Illuminated) CMOS active-pixel digital image sensor with a pixel array of 4208 (H) × 3120 (V) (4224 (H) × 3136 (V) including border pixels). It incorporates sophisticated on-chip camera functions such as mirroring, column and row skip modes, and snapshot mode. It is programmable through a simple two-wire serial interface and has very low power consumption.

The AR1335 digital image sensor features ON Semiconductor's breakthrough low-noise CMOS imaging technology that achieves near-CCD image quality (based on signal-to-noise ratio and low-light sensitivity) while maintaining the inherent size, cost, and integration advantages of CMOS.

The AR1335 sensor can generate full resolution image at up to 30 frames per second (fps). An on-chip analog-to-digital converter (ADC) generates a 10-bit value for each pixel.

**Table 1. KEY PERFORMANCE PARAMETERS**

| Parameter | | Typical Value |
|---|---|---|
| Optical Format | | 1/3.2-inch 13 Mp (4:3) |
| Active Pixels | | 4208 (H) × 3120 (V) |
| Pixel Size | | 1.1 μm Back Side Illuminated (BSI) |
| Chief Ray Ange (CRA) | | 32°: Bare Die<br>11°: Bare Die |
| Die Size | | 6.3 × 5.7 mm |
| Input Clock Frequency | | 6–48 MHz |
| Interface | | 4-lane MIPI (2- and 3-lane Supported); Max Data Rate: 1.2 Gbps/Lane |
| Subsampling Modes (Column and Row) | | skip2, bin2, skip3, bin3, skip4, bin4, skip2bin2 |
| ACD Resolution | | 10 bits, On-die |
| Analog Gain | | 1–7.75× |
| Digital Gain | | Up to 7.98× |
| Scaler | | Adjustable Scalling Up to 8× |
| Temperature Sensor | | 10-bit, Controlled by Two-wire Serial I/F |
| Compression | | DPCM: 10-8-10, 10-6-10 |
| 3D Support | | Frame Rate and Exposure Synchronization |
| Supply Voltage | $V_{AA}$, $V_{AA}$_PIX | 2.7 V (2.6 V < $V_{RIPPLE}$ < 2.9 V) |
| | $V_{DD}$_IO, $V_{DD}$IO_ANA | 1.8 V (1.7 V < $V_{RIPPLE}$ < 1.9 V) |
| | $V_{DD}$, $V_{DD}$_ANA, $V_{DD}$_PLL, $V_{DD}$_PHY | 1.2 V (1.14 V < $V_{RIPPLE}$ < 1.3 V) |
| Power Consumption | | 270 mW at 60°C (Typ.) at 13 Mp 30 fps |
| Responsivity | | 4700 $e^-$/lux-sec |
| $SNR_{MAX}$ | | 37 dB |
| Dynamic Range | | 69 dB |
| Operating Temperature Range (at Junction) – $T_J$ | | –30°C to +70°C |

## Features

- 13 Mp CMOS Sensor with Advanced 1.1 μm Pixel BSI Technology
- Data Interfaces: 2-, 3-, and 4-lane Serial Mobile Industry Processor Interface (MIPI)
- Bit-depth Compression Available for MIPI Interface: 10-8 and 10-6 to Enable Lower Bandwidth Receivers for Full Frame Rate Applications
- 3D Synchronization Controls to Enable Stereo Video Capture
- 6.8 kbits One-time Programmable Memory (OTPM) for Storing Shading Correction Coefficients and Module Information
- Programmable Controls: Gain, Horizontal and Vertical Blanking, Auto Black Level Offset Correction, Frame Size/Rate, Exposure, Left-right and Top-bottom Image Reversal, Window Size, and Panning
- Two On-die Phase-locked Loop (PLL) Oscillators for Super Low Noise Performance
- On-chip Temperature Sensor
- Bayer Pattern Horizontal Down-size Scaler
- Simple Two-wire Fast-mode + Serial Interface
- Low Dark Current
- Interlaced Multi-exposure Readout Enabling High Dynamic Range (HDR) Still and Video Applications

**Features** (continued)

- On-chip Lens Shading Correction
- Support for External Mechanical Shutter
- Support for External LED or Xenon Flash
- Extended Flash Duration Up to Start of Frame Readout

**Applications**

- Cellular Phones
- Digital Still Cameras
- PC Cameras
- PDAs

**Table 2. MODE OF OPERATION AND POWER**

| Mode | Resolution | Readout Configuration | HFOV | FPS | Power Consumption (mW) |
|---|---|---|---|---|---|
| **4:3 SNAPSHOT MODE** | | | | | |
| 13 M Full Resolution | 4208 × 3120 | 13 M Full Mode | 100% | 30 | 270 |
| 13 M Full Resolution | 4208 × 3120 | 13 M Full Mode | 100% | 24 | 250 |
| VGA | 640 × 480 | Crop + Subsampling + Scalling | 61% | 120 | 190 |
| QVGA | 320 × 240 | Crop + Subsampling + Scalling | 30% | 240 | 165 |
| **16:9 VIDEO MODE 30 FPS** | | | | | |
| 4K UHD | 3840 × 2160 | Cropping | 91% | 30 | 230 |
| 4K Cinema | 4096 × 2160 | Cropping | 97% | 30 | 235 |
| 1080p | 1920 × 1080 | Crop + Subsampling + Scalling | 91% | 30 | 160 |
| 1080p LP | 1920 × 1080 | Crop + Subsampling + Scalling | 91% | 30 | 135 |
| 720p | 1280 × 720 | Crop + Subsampling + Scalling | 91% | 30 | 140 |
| **16:9 VIDEO MODE 60 FPS** | | | | | |
| 1080p | 1920 × 1080 | Crop + Subsampling + Scalling | 91% | 60 | 210 |
| 1080p LP | 1920 × 1080 | Crop + Subsampling + Scalling | 91% | 60 | 180 |
| 720p | 1280 × 720 | Crop + Subsampling + Scalling | 91% | 60 | 175 |
| **3M 30 FPS** | | | | | |
| 3M | 2000 × 1500 | Crop + Subsampling + Scalling | 95% | 30 | 195 |
| 3M LP | 2000 × 1500 | Crop + Subsampling + Scalling | 95% | 30 | 170 |
| **16:9 VIDEO MODE 120 FPS** | | | | | |
| 720p | 1280 × 720 | Crop + Subsampling + Scalling | 91% | 120 | 260 |

## ORDERING INFORMATION

**Table 3. ORDERING INFORMATION – AVAILABLE PART NUMBERS**

| Pad Number | Product Description | Orderable Product Attribute Description |
|---|---|---|
| AR1335CSSC11SMD20 | 11° CRA, 200 μm Bare Die | |
| AR1335CSSC32SMD20 | 32° CRA, 200 μm Bare Die | |
| AR1335CSSC32SMFAH−GEVB | Head Board | |

See the ON Semiconductor *Device Nomenclature* document (TND310/D) for a full description of the naming convention used for image sensors. For reference documentation, including information on evaluation kits, please visit our web site at www.onsemi.com.

## FUNCTION OVERVIEW

The AR1335 is a progressive-scan sensor that generates a stream of pixel data at a constant frame rate. It uses an on-chip, phase-locked loop (PLL) to generate all internal clocks from a single master input clock running between 6 and 48 MHz. The maximum data rate is 1.2 Gbps per lane. A block diagram of the sensor is shown in Figure 1.

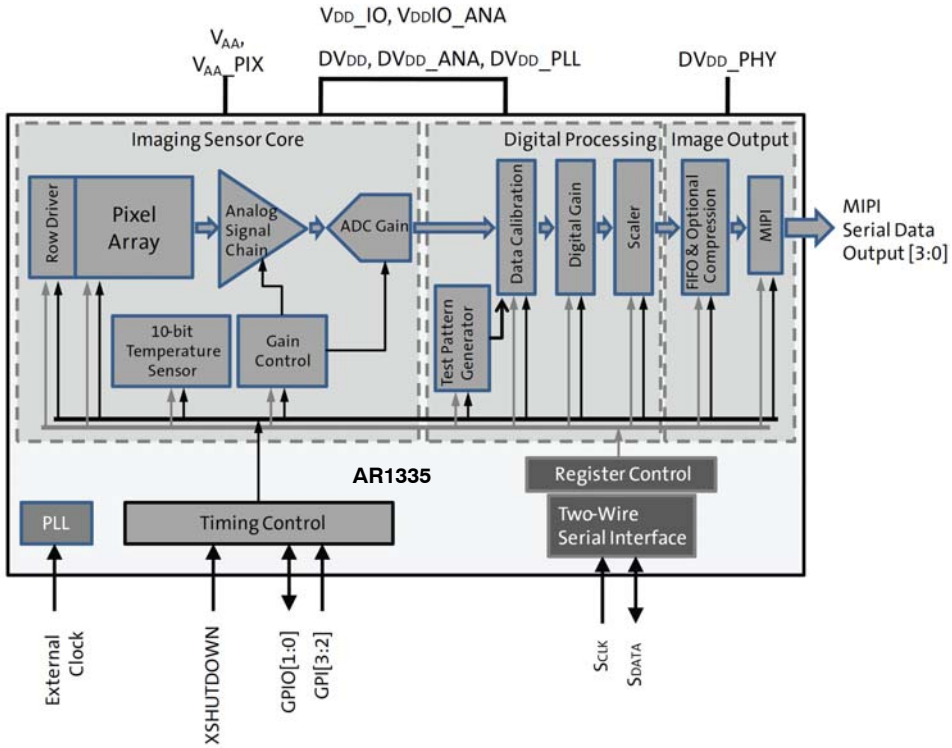



**Figure 1. Block Diagram**

The core of the sensor is a 13 Mp active-pixel array. The timing and control circuitry sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and reading that row, the pixels in the row integrate incident light. The exposure is controlled by varying the time interval between reset and readout. Once a row has been read, the data from the columns is sequenced through an analog signal chain (providing offset correction and gain), and then through an ADC. The output from the ADC is a 10-bit value for each pixel in the array. The ADC output passes through a digital processing signal chain (which provides further data path corrections and applies digital gain).

The pixel array contains optically active and light-shielded (“dark”) pixels. The dark pixels are used to provide data for on-chip offset-correction algorithms (“black level” control).

The sensor contains a set of control and status registers that can be used to control many aspects of the sensor behavior including the frame size, exposure, and gain setting. These registers can be accessed through a two-wire serial interface.

The output from the sensor is a Bayer pattern; alternate rows are a sequence of either green and red pixels or blue and green pixels. The offset and gain stages of the analog signal chain provide global per-color control of the pixel data.

The control registers, timing and control, and digital processing functions shown in Figure 1 are partitioned into three logical parts:

- A sensor core that provides array control and data path corrections. The output of the sensor core is a 10-bit serial pixel data stream qualified by a 4-lane MIPI output clock.
- Data processing functions, including a digital shading correction block to compensate for brightness shading introduced by the lens or chief ray angle (CRA) curve mismatch, digital gain, and dynamic defect correction.
- Output processing functions, including a horizontal scaler, a limiter, a data compressor, an output FIFO, and a serializer.

The output FIFO is present to prevent data bursts by keeping the data rate continuous.

Programmable slew rates are also available to reduce the effect of electromagnetic interference from the output interface.

A flash output signal is provided to allow an external xenon or LED light source to synchronize with the sensor exposure time. Additional I/O signals support the provision of an external mechanical shutter.

## PIXEL ARRAY

The sensor core uses a Bayer color pattern, as shown in Figure 2. The even-numbered rows contain green and red pixels; odd-numbered rows contain blue and green pixels.

Even-numbered columns contain green and blue pixels; odd-numbered columns contain red and green pixels.

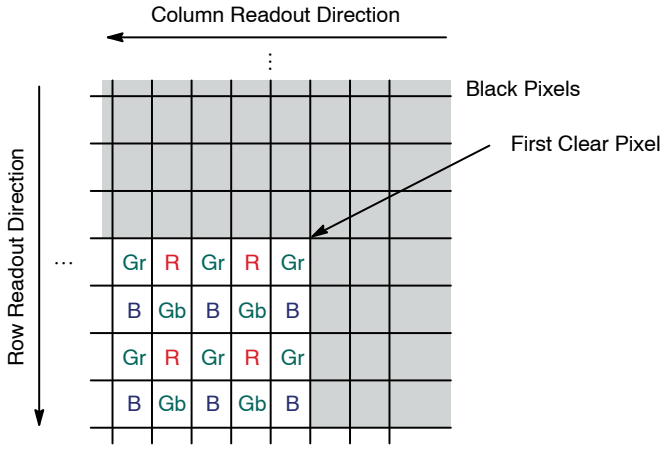



**Figure 2. Pixel Color Pattern Detail (Top Right Corner)**

## OPERATING MODES

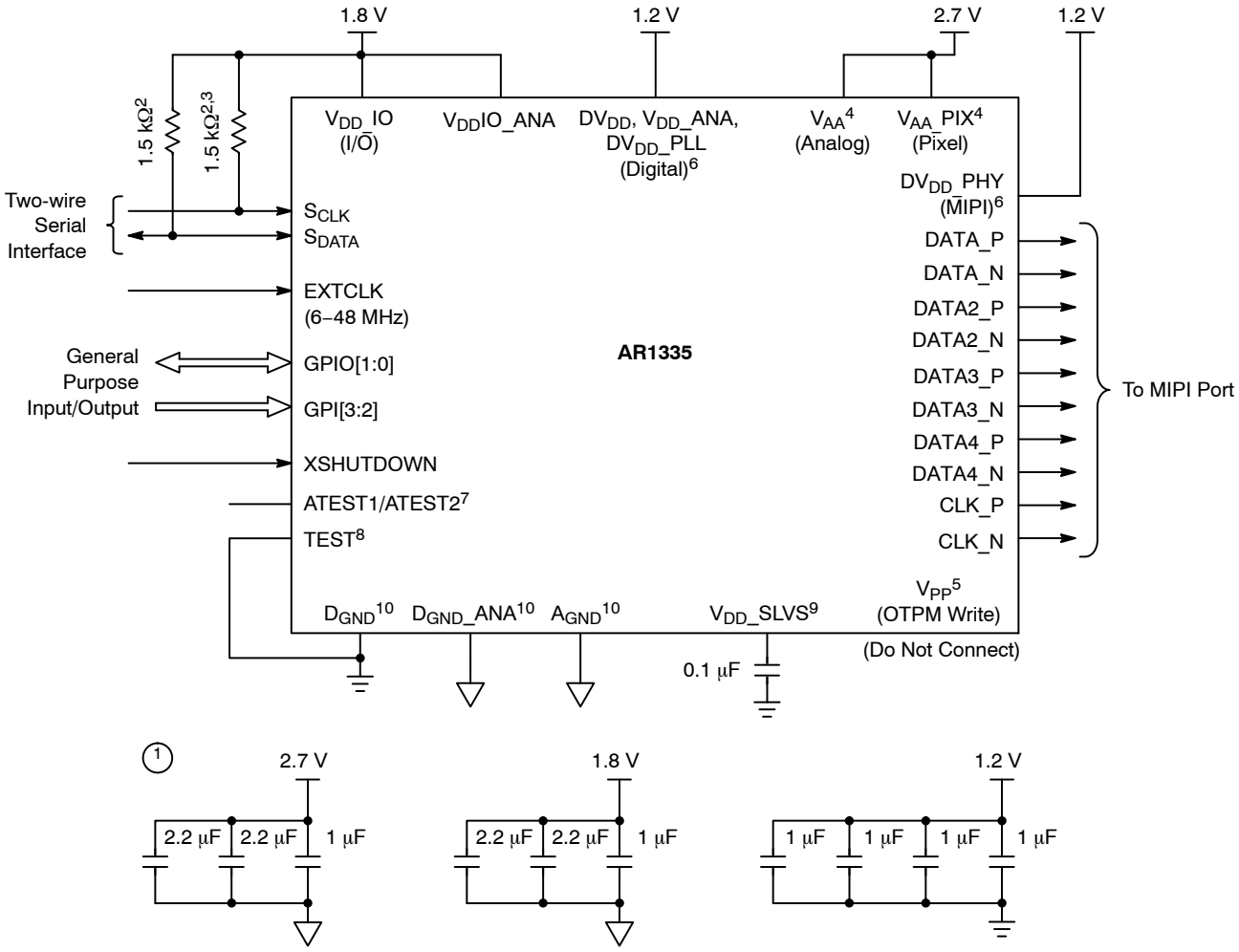
The AR1335 supports MIPI serial output, which can be configured in 2-, 3-, and 4-lanes. There is no parallel data output port. Typical configurations are shown in Figure 3. These operating modes are described in “Control of the Signal Interface”.

For low-noise operation, the AR1335 requires separate power supplies for analog and digital. Incoming digital and analog ground conductors can be tied together next to the die. Both power supply rails should be decoupled from ground using capacitors as close as possible to the die.

**WARNING:** ON Semiconductor does not recommend the use of inductance filters on the power supplies or output signals.

## TYPICAL CONNECTIONS

Figure 3 shows typical AR1335 connections.




Notes:

1. All power supplies should be adequately decoupled; recommended minimum cap values are:
   – 2.7 V: 2 × 2 µF and 1.0 µF
   – 1.2 V: 4 × 1.0 µF
   – 1.8 V: 2 × 2 µF and 1.0 µF
2. Resistor value 1.5 kΩ is recommended, but may be greater for slower two-wire speed.
3. This pull-up resistor is not required if the controller drives a valid logic level on $S_{CLK}$ at all times.
4. $V_{AA}$ and $V_{AA}$_PIX must be tied together.
5. Internal charge pump is used for OTPM programming.
6. Digital and MIPI supply can be tied together.
7. ATEST1/ATEST2 must be left floating.
8. TEST pin must be tied to $D_{GND}$.
9. $V_{DD}$_SLVS must be connected to $D_{GND}$ through a bypass cap (0.1 µF).
10. Analog and digital ground domains can be combined in applications where limited routing resources would lead to inefficient layout and high ground trace impedance if split domains were used.

**Figure 3. Typical Connections**

## SIGNAL DESCRIPTIONS

Table 4 provides signal descriptions for AR1335 die. For pad location and aperture information, refer to the AR1335 die data sheet.

**Table 4. SIGNAL DESCRIPTIONS**

| Name | Type | Description |
|---|---|---|
| EXTCLK | Input | Master clock input, 6–48 MHz |
| XSHUTDOWN | Input | Asynchronous active LOW reset. This pin will turn off all power domains and is the lowest power state of the sensor. When asserted, data output stops and when de-asserted all internal registers are restored to their factory default settings. |
| $S_{CLK}$ | Input | Serial clock for access to control and status registers |
| GPI[3:2] | Input | General purpose inputs. After reset, these pads are powered-down by default; this means that it is not necessary to bond to these pads. These pads can be configured to provide hardware control of: GPI[2]: $S_{ADDR}$, Trigger signal for slave mode and standby. GPI[3]: 3D daisy chain communication input and all options in GPI[2]. ON Semiconductor recommends that unused GPI pins be tied to $D_{GND}$, but can also be left floating. |
| GPIO[1:0] | I/O | General purpose inputs and outputs. After reset, these pads are not powered-down since its default use is as output. These pads can be configured to provide hardware control of: GPIO[0]: Flash output (default),all input options in GPI[2]. GPIO[1]: Shutter output (default), 3D daisy chain communication output and all options in GPI[2]. ON Semiconductor recommends that unused GPIO pins be tied to $D_{GND}$, but they can also be left floating. |
| TEST | Input | TEST must be tied to DGND for normal operation |
| $S_{DATA}$ | I/O | Serial data from reads and writes to control and status registers |
| DATA_P | Output | Differential MIPI (sub-LVDS) serial data 1st lane (positive) |
| DATA_N | Output | Differential MIPI (sub-LVDS) serial data 1st lane (negative) |
| DATA2_P | Output | Differential MIPI (sub-LVDS) serial data 2nd lane (positive) |
| DATA2_N | Output | Differential MIPI (sub-LVDS) serial data 2nd lane (negative) |
| DATA3_P | Output | Differential MIPI (sub-LVDS) serial data 3rd lane (positive) |
| DATA3_N | Output | Differential MIPI (sub-LVDS) serial data 3rd lane (negative) |
| DATA4_P | Output | Differential MIPI (sub-LVDS) serial data 4th lane (positive) |
| DATA4_N | Output | Differential MIPI (sub-LVDS) serial data 4th lane (negative) |
| CLK_P | Output | Differential MIPI (sub-LVDS) serial clock/strobe (positive). |
| CLK_N | Output | Differential MIPI (sub-LVDS) serial clock/strobe (negative) |
| $V_{PP}$ | Supply | Do not connect |
| $DV_{DD}$_PLL | Supply | PLL power supply |
| $DV_{DD}$_PHY | Supply | Digital PHY power supply. Digital power supply for the serial interface (1.2 V) |
| $V_{AA}$ | Supply | Analog power supply (2.7 V) |
| $V_{AA}$_PIX | Supply | Analog power supply for the pixel array (2.7 V) |
| $A_{GND}$ | Supply | Analog ground |
| $DV_{DD}$, $DV_{DD}$_ANA | Supply | 1.2 V digital power supply inputs |
| $V_{DD}$_IO, $V_{DD}$IO_ANA | Supply | I/O power supply (1.8 V) |
| $D_{GND}$ | Supply | Common ground for digital and I/O |
| $V_{DD}$_SLVS | Supply | The $V_{DD}$_SLVS is an internally driven 0.4 V reference voltage. It must connect to $D_{GND}$ through a bypass cap (0.1 μF). |

## OUTPUT DATA FORMAT

### Serial Pixel Data Interface

The AR1335 serial pixel data interface implements data/clock and data/strobe signaling in accordance with the MIPI specifications. The RAW10, RAW8, and RAW6 image data formats are supported.

## TWO-WIRE SERIAL REGISTER INTERFACE

The two-wire serial interface bus enables read/write access to control and status registers within the AR1335. The interface protocol uses a master/slave model in which a master controls one or more slave devices. The sensor acts as a slave device. The master generates a clock ($S_{CLK}$) that is an input to the sensor and is used to synchronize transfers. Data is transferred between the master and the slave on a bidirectional signal ($S_{DATA}$). $S_{DATA}$ is pulled up to $V_{DD}$ off-chip by a 1.5 kΩ resistor. Either the slave or master device can drive $S_{DATA}$ LOW – the interface protocol determines which device is allowed to drive $S_{DATA}$ at any given time.

The protocols described in the two-wire serial interface specification allow the slave device to drive $S_{CLK}$ LOW; the AR1335 uses $S_{CLK}$ as an input only and therefore never drives it LOW.

### Protocol

Data transfers on the two-wire serial interface bus are performed by a sequence of low-level protocol elements:

1. A (Repeated) Start Condition
2. A Slave Address/Data Direction Byte
3. An (a no) Acknowledge Bit
4. A Message Byte
5. A Stop Condition

The bus is idle when both $S_{CLK}$ and $S_{DATA}$ are HIGH. Control of the bus is initiated with a start condition, and the bus is released with a stop condition. Only the master can generate the start and stop conditions.

*Start Condition*

A start condition is defined as a HIGH-to-LOW transition on $S_{DATA}$ while $S_{CLK}$ is HIGH. At the end of a transfer, the master can generate a start condition without previously generating a stop condition; this is known as a “repeated start” or “restart” condition.

*Stop Condition*

A stop condition is defined as a LOW-to-HIGH transition on $S_{DATA}$ while $S_{CLK}$ is HIGH.

*Data Transfer*

Data is transferred serially, 8 bits at a time, with the MSB transmitted first. Each byte of data is followed by an acknowledge bit or a no-acknowledge bit. This data transfer mechanism is used for the slave address/data direction byte and for message bytes.

One data bit is transferred during each $S_{CLK}$ clock period. $S_{DATA}$ can change when $S_{CLK}$ is LOW and must be stable while $S_{CLK}$ is HIGH.

*Slave Address/Data Direction Byte*

Bits [7:1] of this byte represent the device slave address and bit [0] indicates the data transfer direction. A “0” in bit [0] indicates a WRITE, and a “1” indicates a READ. The default slave addresses used by the AR1335 for the MIPI-configured sensor are 0x6C (write address) and 0x6D (read address) in accordance with the MIPI specification. Alternate slave addresses of 0x6E(write address) and 0x6F(read address) can be selected by enabling and asserting the $S_{ADDR}$ signal through the GPI pad.

An alternate slave address can also be programmed through R0x31FC.

*Message Byte*

Message bytes are used for sending register addresses and register write data to the slave device and for retrieving register read data.

*Acknowledge Bit*

Each 8-bit data transfer is followed by an acknowledge bit or a no-acknowledge bit in the $S_{CLK}$ clock period following the data transfer. The transmitter (which is the master when writing, or the slave when reading) releases $S_{DATA}$. The receiver indicates an acknowledge bit by driving $S_{DATA}$ LOW. As for data transfers, $S_{DATA}$ can change when $S_{CLK}$ is LOW and must be stable while $S_{CLK}$ is HIGH.

*No-Acknowledge Bit*

The no-acknowledge bit is generated when the receiver does not drive $S_{DATA}$ LOW during the $S_{CLK}$ clock period following a data transfer. A no-acknowledge bit is used to terminate a read sequence.

### Typical Sequence

A typical READ or WRITE sequence begins by the master generating a start condition on the bus. After the start condition, the master sends the 8-bit slave address/data direction byte. The last bit indicates whether the request is for a read or a write, where a “0” indicates a write and a “1” indicates a read. If the address matches the address of the slave device, the slave device acknowledges receipt of the address by generating an acknowledge bit on the bus.

If the request was a WRITE, the master then transfers the 16-bit register address to which the WRITE should take place. This transfer takes place as two 8-bit sequences and the slave sends an acknowledge bit after each sequence to indicate that the byte has been received. The master then transfers the data as an 8-bit sequence; the slave sends an acknowledge bit at the end of the sequence. The master stops writing by generating a (re)start or stop condition.

If the request was a READ, the master sends the 8-bit write slave address/data direction byte and 16-bit register address, the same way as with a WRITE request. The master then generates a (re)start condition and the 8-bit read slave address/data direction byte, and clocks out the register data, eight bits at a time. The master generates an acknowledge bit after each 8-bit transfer. The slave's internal register address is automatically incremented after every 8 bits are transferred. The data transfer is stopped when the master sends a no-acknowledge bit.

### Single READ from Random Location

This sequence (Figure 4) starts with a dummy WRITE to the 16-bit address that is to be used for the READ. The master terminates the WRITE by generating a restart condition. The master then sends the 8-bit read slave address/data direction byte and clocks out one byte of register data. The master terminates the READ by generating a no-acknowledge bit followed by a stop condition. Figure 4 shows how the internal register address maintained by the AR1335 is loaded and incremented as the sequence proceeds.

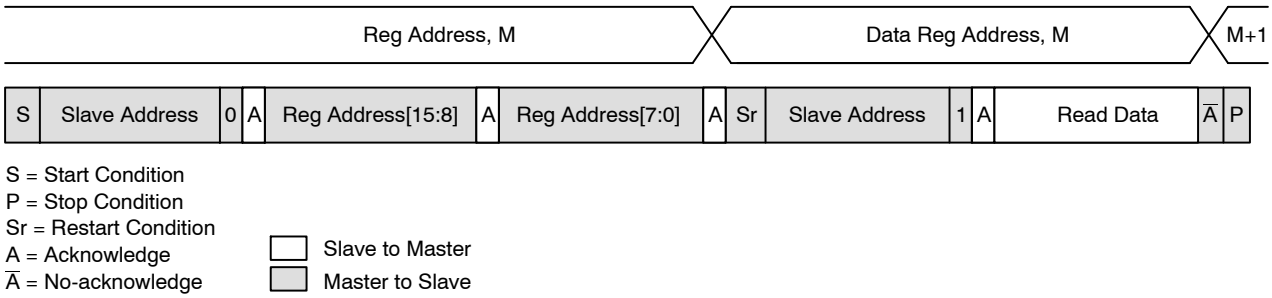



**Figure 4. Single READ from Random Location**

### Single READ from Current Location

This sequence (Figure 5) performs a read using the current value of the AR1335 internal register address. The master terminates the READ by generating a no-acknowledge bit followed by a stop condition. The figure shows two independent READ sequences.

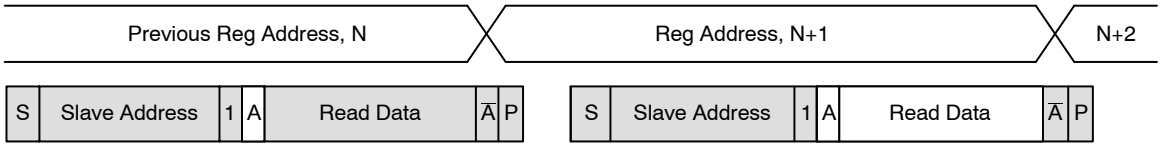



**Figure 5. Single READ from Current Location**

### Sequential READ, Start from Random Location

This sequence (Figure 6) starts in the same way as the single READ from random location (Figure 4). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit and continues to perform byte READs until "L" bytes have been read.

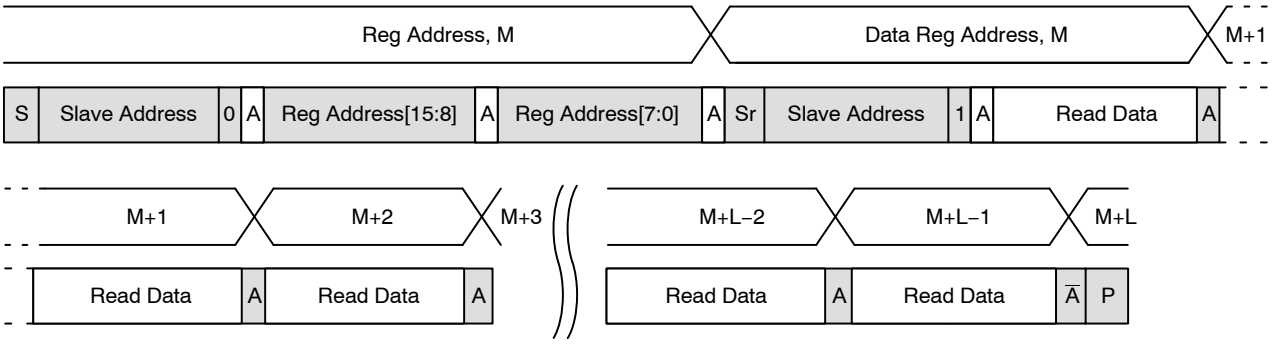



**Figure 6. Sequential READ, Start from Random Location**

### Sequential READ, Start from Current Location

This sequence (Figure 7) starts in the same way as the single READ from current location (Figure 5). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit and continues to perform byte READs until “L” bytes have been read.


**Figure 7. Sequential READ, Start from Current Location**

### Single WRITE to Random Location

This sequence (Figure 8) begins with the master generating a start condition. The slave address/data direction byte signals a WRITE and is followed by the HIGH then LOW bytes of the register address that is to be written. The master follows this with the byte of write data. The WRITE is terminated by the master generating a stop condition.

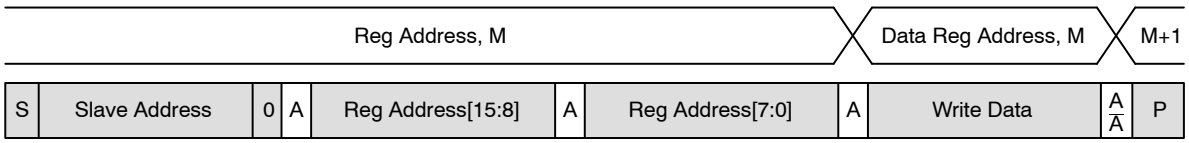



**Figure 8. Single WRITE to Random Location**

### Sequential WRITE, Start at Random Location

This sequence (Figure 9) starts in the same way as the single WRITE to random location (Figure 8). Instead of generating a no-acknowledge bit after the first byte of data has been transferred, the master generates an acknowledge bit and continues to perform byte WRITEs until “L” bytes have been written. The WRITE is terminated by the master generating a stop condition.

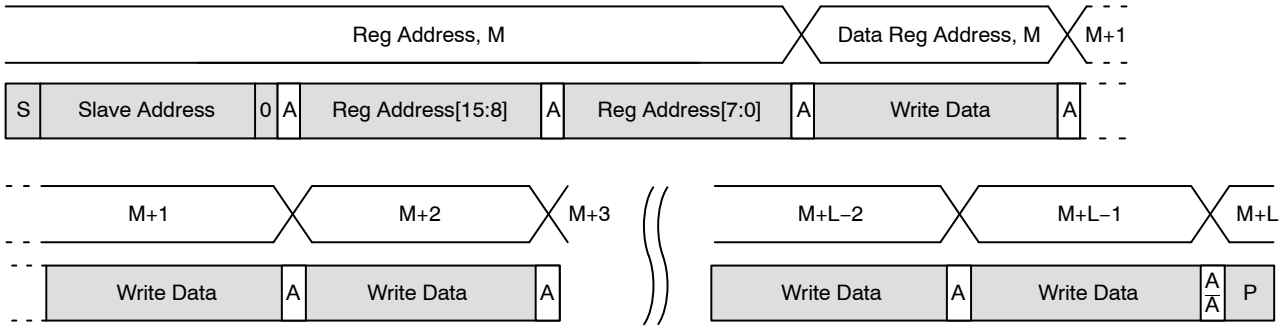



**Figure 9. Sequential WRITE, Start at Random Location**

## REGISTERS

The AR1335 provides a 16-bit register address space accessed through a serial interface (“Two-Wire Serial Register Interface”). Each register location is 8 or 16 bits in size.

The address space is divided into the five major regions shown in Table 5. The remainder of this section describes these registers in detail.

**Table 5. ADDRESS SPACE REGIONS**

| Address Range | Description |
|---|---|
| 0x0000–0x0FFF | Configuration registers (read-only and read-write dynamic registers) |
| 0x1000–0x1FFF | Parameter limit registers (read-only static registers) |
| 0x2000–0x2FFF | Image statistics registers (none currently defined) |
| 0x3000–0x3FFF | Manufacturer-specific registers (read-only and read-write dynamic registers) |
| 0x4000–0xFFFF | Reserved (undefined) |

### Register Notation

The underlying mechanism for reading and writing registers provides byte write capability. However, it is convenient to consider some registers as multiple adjacent bytes. The AR1335 uses 8-bit, 16-bit, and 32-bit registers, all implemented as 1 or more bytes at naturally aligned, contiguous locations in the address space.

In this document, registers are described either by address or by name. When registers are described by address, the size of the registers is explicit. For example, R0x3024 is an 8-bit register at address 0x3024, and R0x3000–1 is a 16-bit register at address 0x3000–0x3001. When registers are described by name, the size of the register is implicit. It is necessary to refer to the register table to determine that model_id is a 16-bit register.

*Register Aliases*

A consequence of the internal architecture of the AR1335 is that some registers are decoded at multiple addresses. Some registers in “configuration space” are also decoded in “manufacturer-specific space”. To provide unique names for all registers, the name of the register within manufacturer-specific register space has a trailing underscore. For example, R0x0000–1 is model_id, and R0x3000–1 is model_id_. The effect of reading or writing a register through any of its aliases is identical.

*Bit Fields*

Some registers provide control of several different pieces of related functionality, and this makes it necessary to refer to bit fields within registers. As an example of the notation used for this, the least significant 4 bits of the model_id register are referred to as model_id[3:0] or R0x0000–1[3:0].

*Bit Field Aliases*

In addition to the register aliases described above, some register fields are aliased in multiple places. For example, R0x0100 (mode_select) has only one operational bit, R0x0100[0]. This bit is aliased to R0x301A–B[2]. The effect of reading or writing a bit field through any of its aliases is identical.

*Byte Ordering*

Registers that occupy more than one byte of address space are shown with the lowest address in the highest-order byte lane to match the byte-ordering on the data bus. For example, the model_id register is R0x0000–1. In the register table the default value is shown as 0x0153. This means that a read from address 0x0000 would return 0x01, and a read from address 0x0001 would return 0x53. When reading this register as two 8-bit transfers on the serial interface, the 0x01 will appear on the serial interface first, followed by the 0x53.

*Address Alignment*

All register addresses are aligned naturally. Registers that occupy 2 bytes of address space are aligned to even 16-bit addresses, and registers that occupy 4 bytes of address space are aligned to 16-bit addresses that are an integer multiple of 4.

*Bit Representation*

For clarity, 32-bit hex numbers are shown with an underscore between the upper and lower 16 bits. For example: 0x3000_01AB.

*Data Format*

Most registers represent an unsigned binary value or set of bit fields. For all other register formats, the format is stated explicitly at the start of the register description. The notation for these formats is shown in Table 6.

**Table 6. DATA FORMATS**

| Name | Description |
|---|---|
| $FIX_{16}$ | Signed fixed-point, 16-bit number: two's complement number, 8 fractional bits.<br>Examples:<br>0x0100 = 1.0,<br>0x8000 = –128,<br>0xFFFF = –0.0039065 |
| $UFIX_{16}$ | Unsigned fixed-point, 16-bit number: 8.8 format.<br>Examples:<br>0x0100 = 1.0,<br>0x280 = 2.5 |
| $FLP_{32}$ | Signed floating-point, 32-bit number: IEEE 754 format.<br>Example:<br>0x4280_0000 = 64.0 |

## Register Behavior

Registers vary from "read-only", "read/write", and "read, write-1-to-clear".

*Double-Buffered Registers*

Some sensor settings cannot be changed during frame readout. For example, changing R0x0344–5 (x_addr_start) partway through frame readout would result in inconsistent row lengths within a frame. To avoid this, the AR1335 double-buffers many registers by implementing a "pending" and a "live" version. Reads and writes access the pending register. The live register controls the sensor operation.

The value in the pending register is transferred to a live register at a fixed point in the frame timing, called frame start. Frame start is defined as the point at which the first dark row is read out internally to the sensor. In the register tables the "Frame Sync'd" column shows which registers or register fields are double-buffered in this way.

*Using Ground_parameter_hold*

Register grouped_parameter_hold (R0x0104) can be used to inhibit transfers from the pending to the live registers. When the AR1335 is in streaming mode, this register should be written to "1" before making changes to any group of registers where a set of changes is required to take effect simultaneously. When this register is written to "0", all transfers from pending to live registers take place on the next frame start.

An example of the consequences of failing to set this bit follows:

An external auto exposure algorithm might want to change both gain and integration time between two frames. If the next frame starts between these operations, it will have the new gain, but not the new integration time, which would return a frame with the wrong brightness that might lead to a feedback loop with the AE algorithm resulting in flickering.

*Bad Frames*

A bad frame is a frame where all rows do not have the same integration time or where offsets to the pixel values have changed during the frame.

Many changes to the sensor register settings can cause a bad frame. For example, when line_length_pck (R0x0342–3) is changed, the new register value does not affect sensor behavior until the next frame start. However, the frame that would be read out at that frame start will have been integrated using the old row width, so reading it out using the new row width would result in a frame with an incorrect integration time.

By default, bad frames are masked. If the masked bad frame option is enabled, both LV and FV are inhibited for these frames so that the vertical blanking time between frames is extended by the frame time.

In the register tables, the "Bad Frame" column shows where changing a register or register field will cause a bad frame. This notation is used:

- *N* – No. Changing the register value will not produce a bad frame.
- *Y* – Yes. Changing the register value might produce a bad frame.
- *YM* – Yes; but the bad frame will be masked out when mask_corrupted_frames (R0x0105) is set to "1."

*Changes to Integration Time*

If the integration time is changed while FV is asserted for frame *n*, the first frame output using the new integration time is frame *(n + 2)*. The sequence is as follows:

1. During frame *n*, the new integration time is held in the pending register.
2. At the start of frame (*n + 1*), the new integration time is transferred to the live register. Integration for each row of frame (*n + 1*) has been completed using the old integration time.
3. The earliest time that a row can start integrating using the new integration time is immediately after that row has been read for frame (*n + 1*).
   The actual time that rows start integrating using the new integration time is dependent upon the new value of the integration time.
4. When frame (*n + 2*) is read out, it will have been integrated using the new integration time.

If the integration time is changed on successive frames, each value written will be applied for a single frame; the latency between writing a value and it affecting the frame readout remains at two frames.

*Changes to Gain Settings*

Usually, when the gain settings are changed, the gain is updated on the next frame start. When the integration time and the gain are changed at the same time, the gain update is held off by one frame so that the first frame output with the new integration time also has the new gain applied. In this case, a new gain should not be set during the extra frame delay. There is an option to turn off the extra frame delay by setting R0x301A–B[14].

## READING THE SENSOR REVISION NUMBER

Follow the steps below to obtain the revision number of the image sensor:

1. Set the register bit field R0x301A[5] = 1.
2. Read the register bit fields R0x31FE[3:0].
3. Convert the binary number to decimal to obtain customer revision.

For example, binary value “0001” = sensor revision 1.

## PROGRAMMING RESTRICTIONS

Table 7 shows a list of programming rules that must be adhered to for correct operation of the AR1335. It is recommended that these rules are encoded into the device driver stack – either implicitly or explicitly.

**Table 7. PROGRAMMING RULES**

| Parameter | Minimum Value | Maximum Value |
|---|---|---|
| coarse_integration_time | 8 rows | frame_length_lines – coarse_integration_time_max_margin |
| digital_gain_* | digital_gain_min | digital_gain_max |
| digital_gain_* is an integer multiple of digital_gain_step_size | | |
| line_length_pck | min_line_length_pck | max_line_length_pck |
| | ((x_addr_end – x_addr_start + x_odd_inc) / xskip) + min_line_blanking_pck | |
| | line_length_pck ≥ (x_output_size + constant) × (vt_pix_clk period) / (op_pix_clk period)<br>Note: Constant is 0x20 and 0x68 for MIPI. | |
| frame_length_lines | min_frame_length_lines | max_frame_length_lines |
| | ((y_addr_end – y_addr_start + y_odd_inc) / yskip) + min_frame_blanking_lines | |
| x_addr_start (must be an even number) | x_addr_min | x_addr_max |
| x_addr_end (must be an odd number) | x_addr_start | x_addr_max |
| (x_addr_end – x_addr_start + x_odd_inc) | Must be multiple of 8 for skip1, 16 for skip2, 32 for skip4 | must be positive |
| y_addr_start (must be an even number) | y_addr_min | y_addr_max |
| y_addr_end (must be an odd number) | y_addr_start | y_addr_max |
| (y_addr_end – y_addr_start + y_odd_inc) | Must be multiple of 8 for skip1, 16 for skip2, 32 for skip4 | must be positive |
| x_odd_inc (must be an odd number) | min_odd_inc | max_odd_inc |
| y_odd_inc (must be an odd number) | min_odd_inc | max_odd_inc |
| scale_m | scaler_m_min | scaler_m_max |
| x_output_size (must be even number – this is enforced in hardware) | 320 | 4224 |
| y_output_size (must be even number – this is enforced in hardware) | 2 | frame_length_lines |

### Output Size Restrictions

The design specification imposes the restriction that an output line is a multiple of 32 bits in length. This imposes an additional restriction on the legal values of x_output_size:

- When R0x0112 [7:0] = 10 (RAW10/RAW8 data), x_output_size must be a multiple of 16 (x_output_size[3:0] = 0).

This restriction only applies when the serial pixel data path is in use. It can be met by rounding up x_output_size to an appropriate multiple. Any extra pixels in the output image as a result of this rounding contain undefined pixel data but are guaranteed not to cause false synchronization on the serial data stream.

There is an additional restriction that x_output_size must be small enough such that the output row time (set by x_output_size, the framing and CRC overhead of 12 bytes and the output clock rate) must be less than the row time of the video array (set by line_length_pck and the video timing clock rate).

### Effect of Scaler on Legal Range of Output Sizes

When the scaler is enabled, it is necessary to adjust the values of x_output_size to match the image size generated by the scaler. The AR1335 will operate incorrectly if the x_output_size is significantly larger than the output image.

To understand the reason for this, consider the situation where the sensor is operating at full resolution and the scaler is enabled with a scaling factor of 32 (half the number of pixels in each direction). This situation is shown in Figure 10.

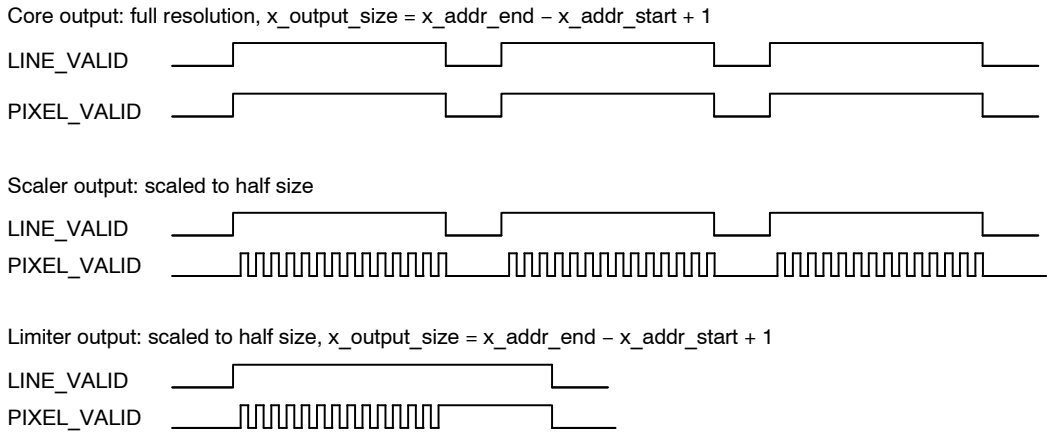



**Figure 10. Effect of Limiter on the Data Path**

In Figure 10, three different stages in the data path (see “Timing Specifications”) are shown. The first stage is the output of the sensor core. The core is running at full resolution and x_output_size is set to match the active array size. The LV signal is asserted once per row and remains asserted for *N* pixel times. The PIXEL_VALID signal toggles with the same timing as LV, indicating that all pixels in the row are valid.

The second stage is the output of the scaler, when the scaler is set to reduce the image size by one-half in x dimension. The effect of the scaler is to combine groups of pixels. Therefore, the row time remains the same, but only half the pixels out of the scaler are valid. This is signaled by transitions in PIXEL_VALID. Overall, PIXEL_VALID is asserted for (*N*/2) pixel times per row.

The third stage is the output of the limiter when the x_output_size is still set to match the active array size. Because the scaler has reduced the amount of valid pixel data without reducing the row time, the limiter attempts to pad the row with (*N*/2) additional pixels. If this has the effect of extending LV across the whole of the horizontal blanking time, the AR1335 will cease to generate output frames.

A correct configuration is shown in Figure 11, in addition to showing the x_output_size reduced to match the output size of the scaler. In this configuration, the output of the limiter does not extend LV.

Figure 11 also shows the effect of the output FIFO, which forms the final stage in the data path. The output FIFO merges the intermittent pixel data back into a contiguous stream. Although not shown in this example, the output FIFO is also capable of operating with an output clock that is at a different frequency from its input clock.

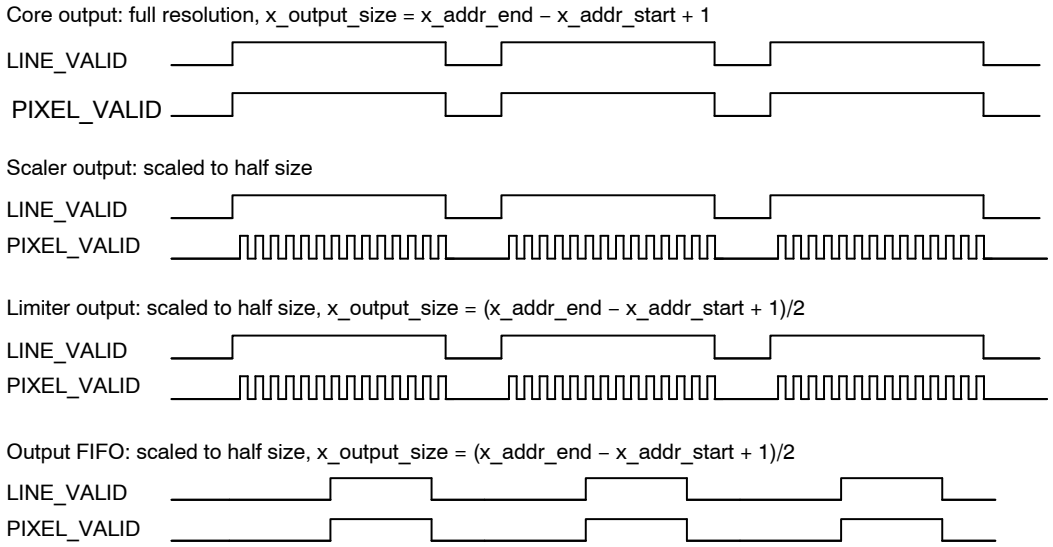



**Figure 11. Timing of Data Path**

### Output Data Timing

The output FIFO acts as a boundary between two clock domains. Data is written to the FIFO in the VT (video timing) clock domain. Data is read out of the FIFO in the OP (output) clock domain.

When the scaler is disabled, the data rate in the VT clock domain is constant and uniform during the active period of each pixel array row readout. When the scaler is enabled, the data rate in the VT clock domain becomes intermittent, corresponding to the data reduction performed by the scaler.

A key constraint when configuring the clock for the output FIFO is that the frame rate out of the FIFO must exactly match the frame rate into the FIFO. When the scaler is disabled, this constraint can be met by imposing the rule that the row time on the serial data stream must be greater than or equal to the row time at the pixel array. The row time on the serial data stream is calculated from the x_output_size and the data_format (8, 10, or 12 bits per pixel), and must include the time taken in the serial data stream for start of frame/row, end of row/frame and checksum symbols.

**WARNING:** If this constraint is not met, the FIFO will either underrun or overrun. FIFO underrun or overrun is a fatal error condition that is signaled through the datapath_status register (R0x306A).

## CONTROL OF THE SIGNAL INTERFACE

This section describes the operation of the signal interface in all functional modes.

The AR1335 sensor provides a MIPI serial interface for pixel data.

### MIPI Serial Pixel Data Interface

The serial pixel data interface uses the following output-only signal pairs:

- DATA_P
- DATA_N
- DATA2_P
- DATA2_N
- DATA3_P
- DATA3_N
- DATA4_P
- DATA4_N
- CLK_P
- CLK_N

The signal pairs use both single-ended and differential signaling, in accordance with the MIPI specification. The serial pixel data interface is enabled by default at power up and after reset.

The DATA_P, DATA_N, DATA2_P, DATA2_N, CLK_P, and CLK_N pads are set to the Ultra Low Power State (ULPS) if the SMIA serial disable bit is asserted (R0x301A−B[12]=1) or when the sensor is in the hardware standby or soft standby system states.

The data_format (R0x0112−3) register can be programmed to the following data format setting:

- 0x0A0A – Sensor supports RAW10 uncompressed data format. This mode is supported by discarding all but the upper 10 bits of a pixel value.
- 0x0808 – Sensor supports RAW8 uncompressed data format. This mode is supported by discarding all but the upper 8 bits of a pixel value.
- 0x0A08 – Sensor supports RAW8 data format in which an adaptive compression algorithm is used to perform 10-bit to 8-bit compression on the upper 10 bits of each pixel value.

The serial_format register (R0x31AE–F) controls which serial interface is in use when the serial interface is enabled (reset_register[12] = 0). The following serial formats are supported:

- 0x0202 – Sensor supports 2-lane MIPI operation
- 0x0203 – Sensor supports 3-lane MIPI operation
- 0x0204 – Sensor supports 4-lane MIPI operation

## SYSTEM STATES

The system states of the AR1335 are represented as a state diagram in Figure 12 and described in subsequent sections. The effect of XSHUTDOWN on the system state and the configuration of the PLL in the different states are shown in Table 8.

The sensor's operation is broken down into three separate states: hardware standby, software standby, and streaming. The transition between these states might take a certain amount of clock cycles as outlined in Table 8.

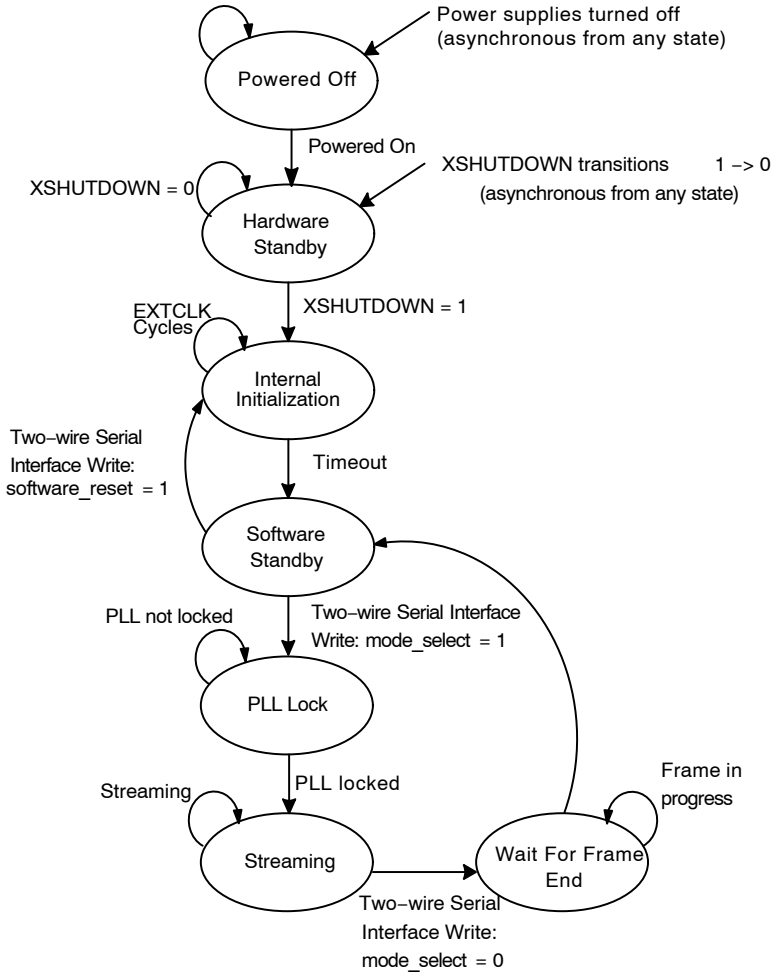



**Figure 12. AR1335 System States**

**Table 8. XSHUTDOWN AND PLL IN SYSTEM STATES**

| State | XSHUTDOWN | PLL |
|---|---|---|
| Powered Off | x | VCO Powered Down |
| Hardware Standby | 0 | |
| Internal Initialization | 1 | |
| Software Standby | | |
| PLL Lock | | VCO Powering Up and Locking, PLL Output Bypassed |
| Streaming | | VCO Running, PLL Output Active |
| Wait for Frame End | | |

## SOFT RESET SEQUENCE

The AR1335 can be reset under software control by writing “1” to software_reset (R0x0103). A software reset asynchronously resets the sensor, truncating any frame that is in progress. The sensor starts the internal initialization sequence, while the PLL and analog blocks are turned off. At this point, the behavior is exactly the same as for the power-on reset sequence.

### Signal State During Reset

Table 9 shows the state of the signal interface during hardware standby (XSHUTDOWN asserted) and the default state during software standby (after exit from hardware standby and before any registers within the sensor have been changed from their default power-up values).

**Table 9. SIGNAL STATE DURING RESET**

| Pad Name | Pad Type | Hardware Standby | Software Standby |
|---|---|---|---|
| EXTCLK | Input | Enabled. Must be driven to a valid logic level. | |
| XSHUTDOWN | Input | Enabled. Must be driven to a valid logic level. | |
| $S_{CLK}$ | Input | Enabled. Must be pulled up or driven to a valid logic level. | |
| $S_{DATA}$ | I/O | Enabled as an input. Must be pulled up or driven to a valid logic level. | |
| GPIO[1:0] | I/O | High-Z. | Logic 0. |
| DATA_P | Output | MIPI: Ultra Low-Power State (ULPS), represented as an LP−00 state on the wire (both wires at 0 V). | |
| DATA_N | Output | | |
| DATA2_P | Output | | |
| DATA2_N | Output | | |
| DATA3_P | Output | | |
| DATA3_N | Output | | |
| DATA4_P | Output | | |
| DATA4_N | Output | | |
| CLK_P | Output | | |
| CLK_N | Output | | |
| GPI[3:2] | Input | Powered down. Can be left disconnected/floating. | |
| TEST | Input | Must be driven to 0 for normal operation. | |

## GENERAL PURPOSE INPUT AND OUTPUT

The AR1335 provides four general purpose input and output pads, as listed in Table 10. GPIO[1:0] are defined as bidirectional (input and output), GPI[3:2] input only.

After power on reset, GPIO[0] is assigned as an output of Flash signal and GPIO[1] is assigned as an output of Shutter signal. Other two GPI are powered down if not used.

**Table 10. GENERAL PURPOSE INPUT AND OUTPUT PAD FUNCTIONS**

| PIN Name | Functions |
|---|---|
| GPIO[0] | General Input and one Output<br>a. (Default Output) Flash<br>b. (Input) All options in GPI2 |
| GPIO[1] | General Input and two Output functions<br>a. (Default Output) Shutter<br>b. (Output) 3-D daisy chain communication output<br>c. (Input) all options in GPI2 |
| GPI[2] | General Input<br>a. SADDR, second $I^2C$ device address<br>b. Trigger signal for Slave Mode<br>c. Standby |
| GPI[3] | General Input<br>a. 3-D daisy chain communication input<br>b. All options in GPI2 |

## STREAMING/STANDBY CONTROL

The AR1335 can be switched between its soft standby and streaming states under pin or register control, as shown in Table 11. Selection of a pin to use for the STANDBY function is described in “General Purpose Input and Output”. The state diagram for transitions between soft standby and streaming states is shown in Figure 12.

**Table 11. STREAMING/STANDBY**

| STANDBY | Streaming R0x301A–B[2] | Description |
|---|---|---|
| Disabled | 0 | Soft Standby |
| Disabled | 1 | Streaming |
| X | 0 | Soft Standby |
| 0 | 1 | Streaming |
| 1 | X | Soft Standby |

## CLOCKING

Default setup provides a physical 220 MHz internal clock for an external input clock of 25 MHz. Maximum allowed for vt_pix_clk is 220 MHz and the design VCO range is between 320–1200 MHz.

The sensor contains two phase-locked loops (PLL). Each PLL contains a prescaler to divide the input clock applied on EXTCLK, a VCO to multiply the prescaler output, and a set of dividers to generate the output clocks.

Note that the data rate of the pixel domain cannot be less than the data rate of the output domain or there will be underflow errors.

Figure 13 shows the different clocks and the names of the registers that contain or are used to control their values.

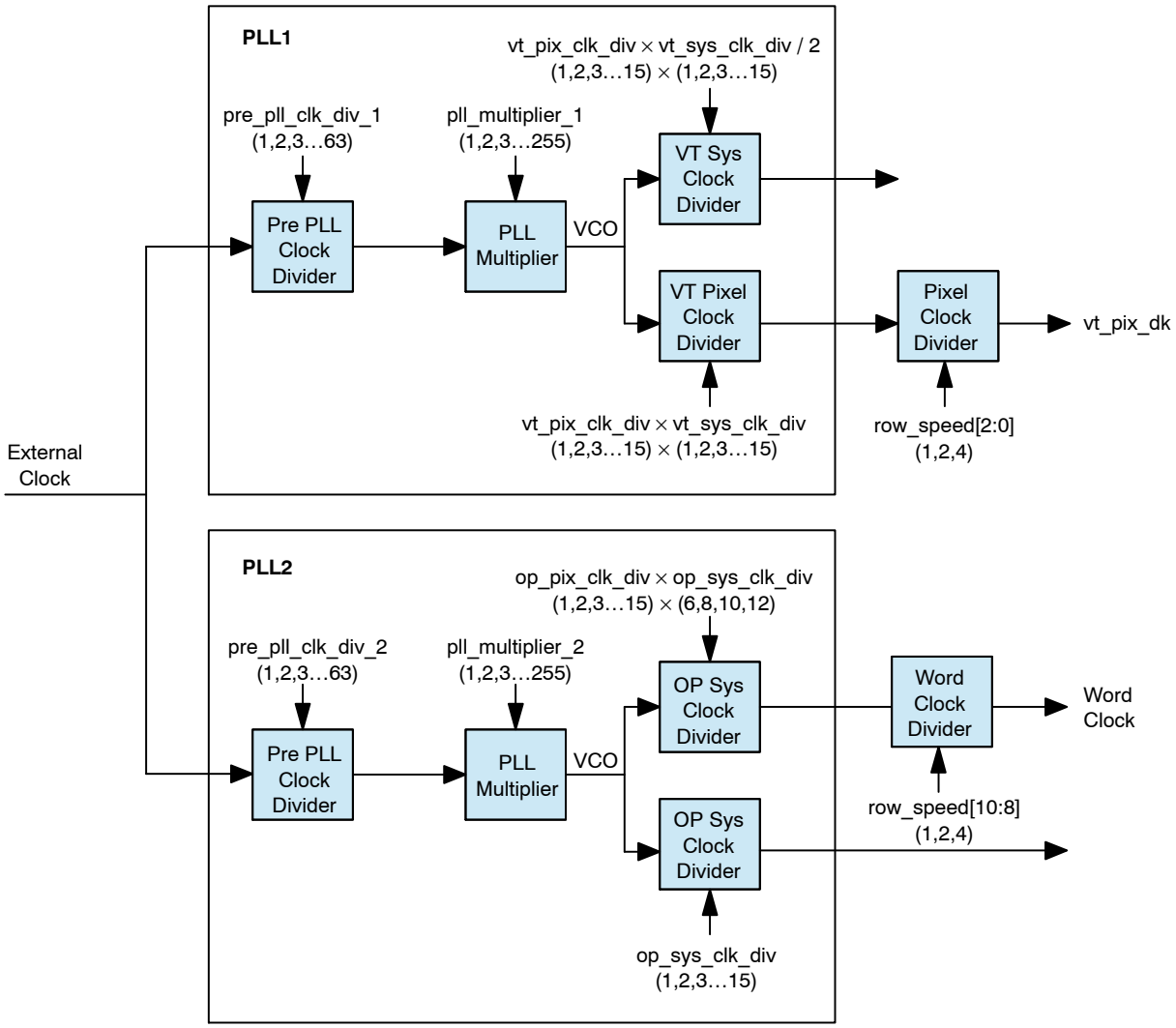



**Figure 13. AR1335 Profile 1/2 Clocking Structure**

Figure 13 shows the different clocks and the names of the registers that contain or are used to control their values. The figure shows the default setting for each divider/multiplier control register and the range of legal values for each divider/multiplier control register.

As per the diagram, the clock frequencies can be calculated as follows:

- pixel_clock (This is the main clock of the sensor). Two data values are processed per clock cycle:

$$clk\_pixel\_freq\_mhz = \frac{ext\_clk\_freq\_mhz \times pll\_multiplier\_1}{pre\_pll\_clk\_div\_1 \times vt\_sys\_clk\_div \times vt\_pix\_clk\_div \times row\_speed[2:0]} \quad \text{(eq. 1)}$$

- word_clock (This is the output domain clock). Two data values are processed per clock cycle:

$$clk\_word\_freq\_mhz = \frac{ext\_clk\_freq\_mhz \times pll\_multiplier\_2}{pre\_pll\_clk\_div\_2 \times op\_sys\_clk\_div \times op\_pix\_clk\_div \times row\_speed[10:8]} \quad \text{(eq. 2)}$$

- Bit_clock (This is the serial domain clock), depending on the number of lanes the data rate will vary:

$$clk\_bit\_freq\_mhz = \frac{ext\_clk\_freq\_mhz \times pll\_multiplier\_2}{pre\_pll\_clk\_div\_2 \times op\_sys\_clk\_div} \quad \text{(eq. 3)}$$

NOTE: In Profile 0, RAW10 data format is required. As a result, op_pix_clk_div must be set to 10. Also, due to the inherent design of the AR1335 sensor, vt_pix_clk_div must be set to 5 for profile 0 mode.)

### PLL Clocking

The PLL divisors should be programmed while the AR1335 is in the software standby state. After programming the divisors, it is necessary to wait for the VCO lock time before enabling the PLL. The PLLs are enabled by entering the streaming state.

The effect of programming the PLL divisors while the AR1335 is in the streaming state is undefined.

*Influence of ccp_data_format*

R0x0112–3 (ccp_data_format) controls whether the pixel data interface will generate 10 or 8 bits or 6 bits per pixel.

When the pixel data interface is generating 10 bits per pixel, op_pix_clk_div must be programmed with the value 10.

## FEATURES

### Shading Correction (SC)

Lenses tend to produce images whose brightness is significantly attenuated near the edges. There are also other factors causing fixed pattern signal gradients in images captured by image sensors. The cumulative result of all these factors is known as image shading. The AR1335 has an embedded shading correction module that can be programmed to counter the shading effects on each individual signal.

*The Correction Function*

Color-dependant solutions are calibrated using the sensor, lens system and an image of an evenly illuminated, featureless gray calibration field. From the resulting image, register values for the correction function (coefficients) can be derived.

The correction functions can then be applied to each pixel value to equalize the response across the image as follows:

$$P_{corrected}(row, col) = P_{sensor}(row, col) \times f(row, col) \quad \text{(eq. 4)}$$

where *P* are the pixel values and *f* is the correction function for each channel.

Each function includes a set of coefficients defined by registers R0x3600–3726. The function's origin is the center point of the function used in the calculation of the coefficients. Using an origin near the central point of symmetry of the sensor response provides the best results. The center point of the function is determined by ORIGIN_C (R0x3782) and ORIGIN_R (R0x3784) and can be used to counter an offset in the system lens from the center of the sensor array.

### One-Time Programmable Memory (OTPM)

The AR1335 features 6.8 kbits of one-time programmable memory (OTPM) for storing shading correction coefficients, individual module, and sensor specific information. The user may program which set to be used. Additional bits are used by the error detection and correction scheme. OTPM can be accessed through two-wire serial interface. The AR1335 uses the auto mode for fast OTPM programming and read operations.

The programming of the OTPM requires the sensor to be fully powered and remain in software standby with its clock input applied. The information will be programmed through the use of the two-wire serial interface, and once the data is written to an internal register, and send a program command to initiate the anti-fusing process. After the sensor has finished programming the OTPM, a status bit will be set to indicate the end of the programming cycle, and the host machine can poll the setting of the status bit through the two-wire serial interface. Only one programming cycle for the 16-bit word can be performed.

Reading the OTPM data requires the sensor to be fully powered and operational with its clock input applied. The data can be read through a register from the two-wire serial interface.

### Image Acquisition Mode

The AR1335 supports an image acquisition mode, the electronic rolling shutter (ERS) mode. This is the normal mode of operation. When the AR1335 is streaming, it generates frames at a fixed rate, and each frame is integrated (exposed) using the ERS. When the ERS is in use, timing and control logic within the sensor sequences through the rows of the array, resetting and then reading each row in turn. In the time interval between resetting a row and subsequently reading that row, the pixels in the row integrate incident light. The integration (exposure) time is controlled by varying the time between row reset and row readout. For each row in a frame, the time between row reset and row readout is fixed, leading to a uniform integration time across the frame. When the integration time is changed (by using the two-wire serial interface to change register settings), the timing and control logic controls the transition from old to new integration time in such a way that the stream of output frames from the AR1335 switches cleanly from the old integration time to the new while only generating frames with uniform integration. See "Changes to Integration Time".

### Window Control

The sequencing of the pixel array is controlled by the x_addr_start, y_addr_start, x_addr_end, and y_addr_end registers. The output image size is controlled by the x_output_size and y_output_size registers.

### Pixel Border

The default settings of the sensor provide a 4208 (H) × 3120 (V) image. A border of up to 8 pixels (4 in subsampling) on each edge can be enabled by reprogramming the x_addr_start, y_addr_start, x_addr_end, y_addr_end, x_output_size, and y_output_size registers accordingly.

## Readout Modes

*Horizontal Mirror*

When the horizontal_mirror bit is set in the image_orientation register, the order of pixel readout within a row is reversed, so that readout starts from x_addr_end and ends at x_addr_start. Figure 14 shows a sequence of 6 pixels being read out with horizontal_mirror = 0 and horizontal_mirror = 1.

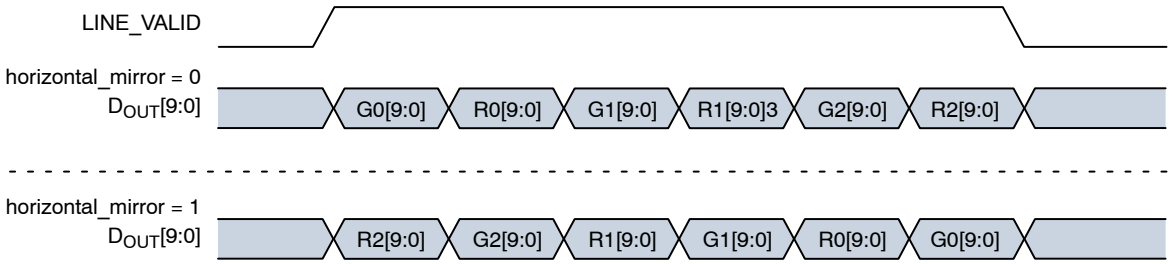



**Figure 14. Effect of horizontal_mirror on Readout Order**

*Vertical Flip*

When the vertical_flip bit is set in the image_orientation register, the order in which pixel rows are read out is reversed, so that row readout starts from y_addr_end and ends at y_addr_start. Figure 15 shows a sequence of 6 rows being read out with vertical_flip = 0 and vertical_flip = 1.

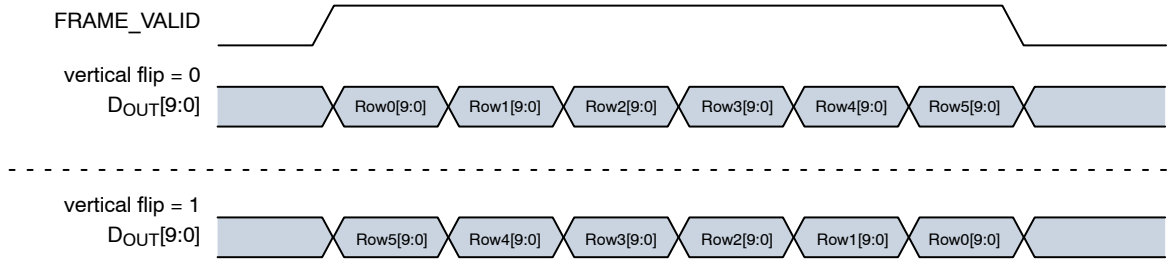



**Figure 15. Effect of vertical_flip on Readout Order**

*Subsampling*

The AR1335 supports subsampling. Subsampling reduces the amount of data processed by the analog signal chain in the AR1335 thereby allowing the frame rate to be increased. Subsampling is enabled by setting x_odd_inc and/or y_odd_inc. Values of 1, 3, 5, and 7 can be supported. Setting both of these variables to 3 reduces the amount of row and column data processed and is equivalent to the $2 \times 2$ skipping readout mode provided by the AR1335 (see Table 12). Setting x_odd_inc = 3 and y_odd_inc = 3 results in a quarter reduction in output image size. Setting x_odd_inc = 5 and y_odd_inc = 5 results in a 1/9 reduction in output image size. Setting x_odd_inc = 7 and y_odd_inc = 7 results in a 1/16 reduction in output image size. Figure 16 shows a sequence of 8 columns being read out with x_odd_inc = 3 and y_odd_inc = 1.

**Table 12. DEFINITIONS FOR PROGRAMMING RULES**

| Name | Definition |
|---|---|
| xskip | xskip = 1 if x_odd_inc = 1<br>xskip = 2 if x_odd_inc = 3<br>xskip = 3 if x_odd_inc = 5<br>xskip = 4 if x_odd_inc = 7 |
| yskip | yskip = 1 if y_odd_inc = 1<br>yskip = 2 if y_odd_inc = 3<br>yskip = 3 if y_odd_inc = 5<br>yskip = 4 if y_odd_inc = 7 |

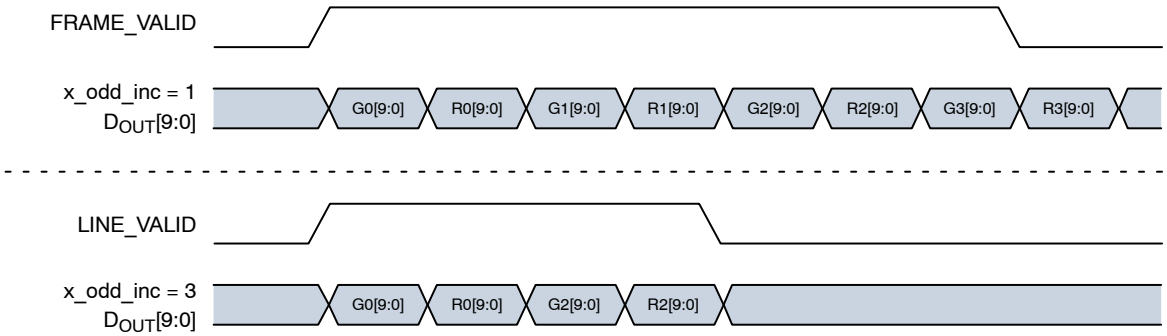



**Figure 16. Effect of x_odd_inc = 3 on Readout Sequence**

A 1/16 reduction in resolution is achieved by setting both x_odd_inc and y_odd_inc to 7. This is equivalent to 4 × 4 skipping readout mode provided by the AR1335. Figure 18 shows a sequence of 16 columns being read out with x_odd_inc = 7 and y_odd_inc = 1.

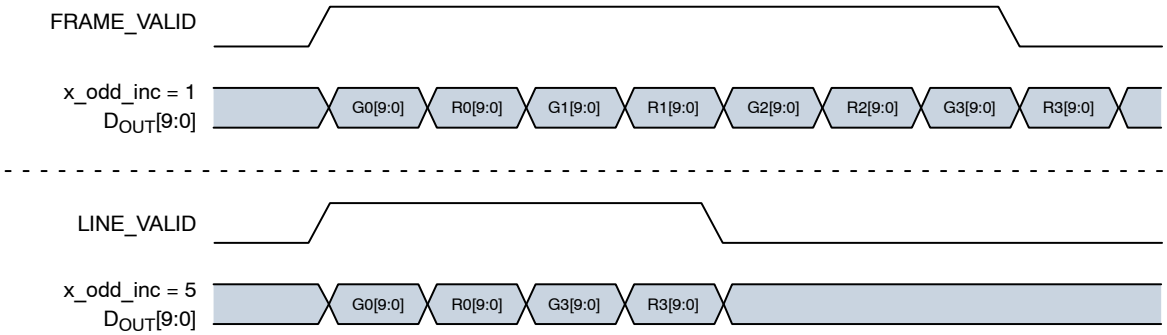



**Figure 17. Effect of x_odd_inc = 5 on Readout Sequence**

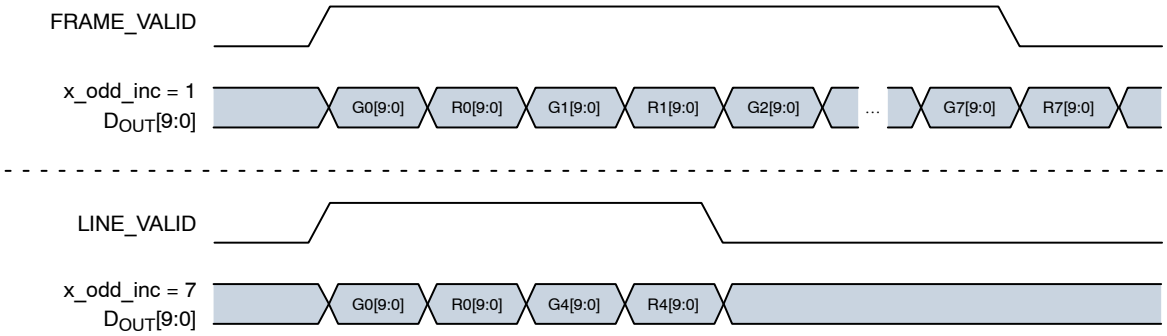



**Figure 18. Effect of x_odd_inc = 7 on Readout Sequence**

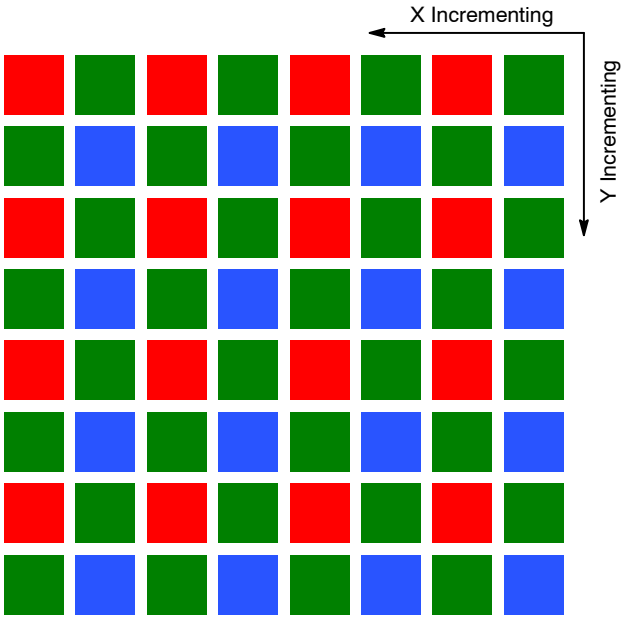



**Figure 19. Pixel Readout (No Subsampling)**

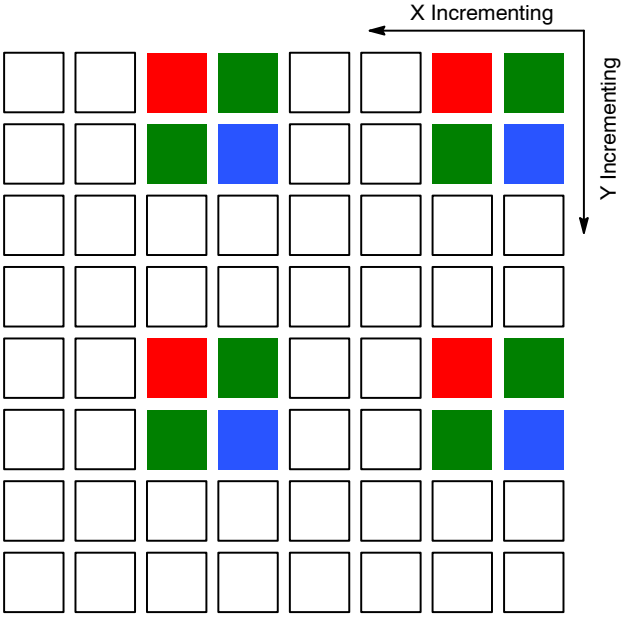



**Figure 20. Pixel Readout (x_odd_inc = 3, y_odd_inc = 3)**

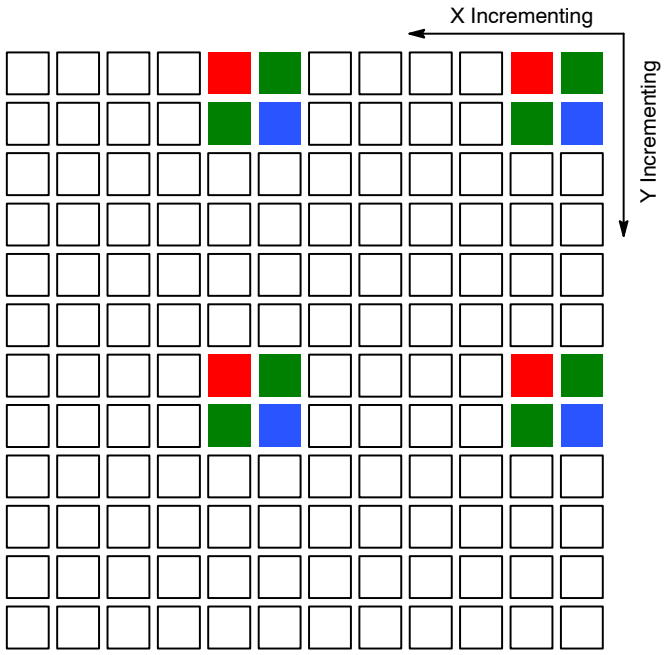



**Figure 21. Pixel Readout (x_odd_inc = 5, y_odd_inc = 5)**

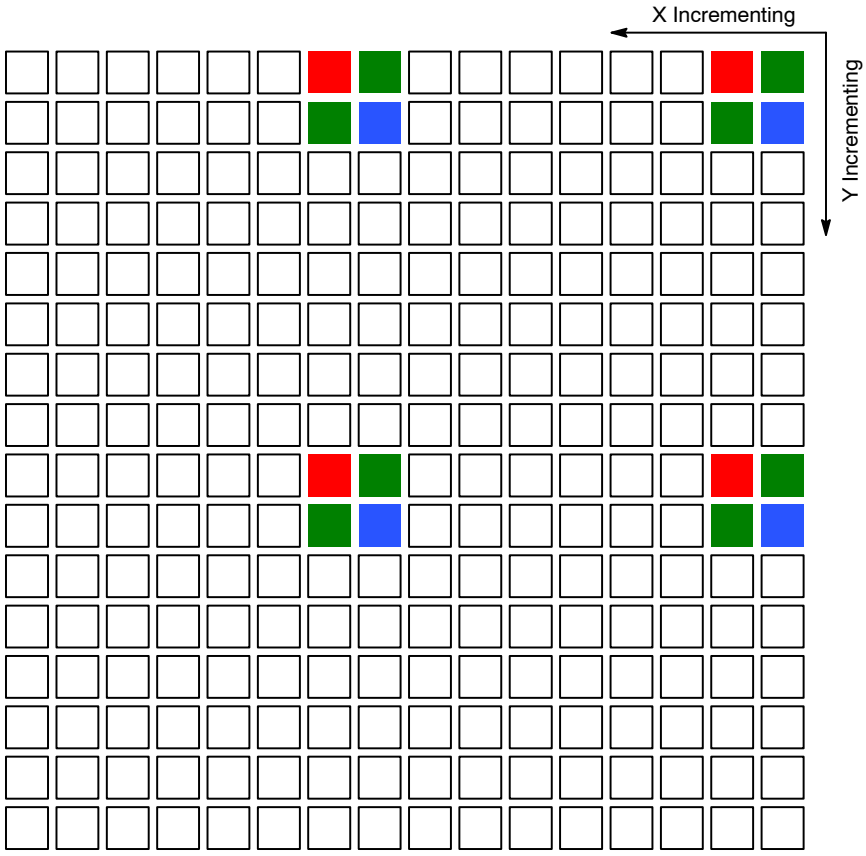



**Figure 22. Pixel Readout (x_odd_inc = 7, y_odd_inc = 7)**

*Programming Restrictions when Sumbsampling*

When subsampling is enabled as a viewfinder mode and the sensor is switched back and forth between full resolution and subsampling, ON Semiconductor recommends that line_length_pck be kept constant between the two modes. This allows the same integration times to be used in each mode.

When subsampling is enabled, it may be necessary to adjust the x_addr_end, x_addr_ start and y_addr_end settings: the values for these registers are required to correspond with rows/columns that form part of the subsampling sequence. The adjustment should be made in accordance with these rules:

x_skip_factor = (x_odd_inc + 1) / 2

y_skip_factor = (y_odd_inc + 1) / 2

- x_addr_start should be a multiple of x_skip_factor × 4
- (x_addr_end − x_addr_start + x_odd_inc) should be a multiple of x_skip_factor x 4
- (y_addr_end − y_addr_start + y_odd_inc) should be a multiple of y_skip_factor x 4

The number of columns/rows read out with subsampling can be found from the equation below:

- columns/rows = (addr_end − addr_start + odd_inc) / skip_factor

Table 13 shows the row or column address sequencing for normal and subsampled readout. In the 2X skip case, there are two possible subsampling sequences (because the subsampling sequence only reads half of the pixels) depending upon the alignment of the start address. Similarly, there will be four possible subsampling sequences in the 4X skip case (though only the first two are shown in Table 13).

**Table 13. ROW ADDRESS SEQUENCING DURING SUBSAMPLING**

| odd_inc = 1 (Normal) | odd_inc = 3 (2x Skip) | odd_inc = 5 (3x Skip) | odd_inc = 7 (4x Skip) |
|---|---|---|---|
| Start = 0 | Start = 0 | Start = 0 | Start = 0 |
| 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |
| 2 | | | |
| 3 | | | |
| 4 | 4 | | |
| 5 | 5 | | |
| 6 | | 6 | |
| 7 | | 7 | |
| 8 | 8 | | 8 |
| 9 | 9 | | 9 |
| 10 | | | |
| 11 | | | |
| 12 | 12 | 12 | |
| 13 | 13 | 13 | |
| 14 | | | |
| 15 | | | |
| 16 | 16 | | 16 |
| 17 | 17 | | 17 |
| 18 | | 18 | |
| 19 | | 19 | |
| 20 | 20 | | |
| 21 | 21 | | |
| 22 | | | |
| 23 | | | |
| 24 | 24 | 24 | 24 |
| 25 | 25 | 25 | 25 |
| 26 | | | |
| 27 | | | |

**Table 13. ROW ADDRESS SEQUENCING DURING SUBSAMPLING** (continued)

| odd_inc = 1 (Normal) | odd_inc = 3 (2x Skip) | odd_inc = 5 (3x Skip) | odd_inc = 7 (4x Skip) |
|---|---|---|---|
| **Start = 0** | **Start = 0** | **Start = 0** | **Start = 0** |
| 28 | 28 | | |
| 29 | 29 | | |
| 30 | | 30 | |
| 31 | | 31 | |
| 32 | 32 | | 32 |
| 33 | 33 | | 33 |
| 34 | | | |
| 35 | | | |
| 36 | 36 | 36 | |
| 37 | 37 | 37 | |
| 38 | | | |
| 39 | | | |
| 40 | 40 | | 40 |
| 41 | 41 | | 41 |
| 42 | | 42 | |
| 43 | | 43 | |
| 44 | 44 | | |
| 45 | 45 | | |
| 46 | | | |
| 47 | | | |

*Scaler*

Scaling is a “zoom out” operation to reduce the size of the output image while covering the same extent as the original image. Each scaled output pixel is calculated by taking a weighted average of a group input pixels which is composed of neighboring pixels.

When compared to skipping, scaling is advantageous because it uses all pixel values to calculate the output image which helps avoid aliasing.

The AR1335 sensor is capable of horizontal scaling only. The scaling factor, programmable in 1/16 steps, is used for horizontal scaling.

The scale factor is determined by:

- *m*, which is adjustable with register R0x0404
- Legal values for m are 16 through 128, giving the user the ability to scale from 1:1 (m = 16) to 1:8 (m = 128).

### Frame Rate Control

The formulas for calculating the frame rate of the AR1335 are shown below.

The line length is programmed directly in pixel clock periods through register line_length_pck. For a specific window size, the minimum line length can be found from in Equation 5:

$$\text{minimum line\_length\_pck} = \left( \frac{\text{x\_addr\_end} - \text{x\_addr\_start} + 1}{\text{subsampling factor}} + \text{min\_line\_blanking\_pck} \right) \quad \text{(eq. 5)}$$

Note that line_length_pck also needs to meet the minimum line length requirement set in register min_line_length_pck. The row time can either be limited by the time it takes to sample and reset the pixel array for each row, or by the time it takes to sample and read out a row. Values for min_line_blanking_pck are provided in “Minimum Row Time”.

The frame length is programmed directly in number of lines in the register frame_line_length. For a specific window size, the minimum frame length can be found in Equation 6:

$$\text{minimum frame\_length\_lines} = \left( \frac{\text{y\_addr\_end} - \text{y\_addr\_start} + 1}{\text{subsampling factor}} + \text{min\_frame\_blanking\_lines} \right) \quad \text{(eq. 6)}$$

The frame rate can be calculated from these variables and the pixel clock speed as shown in Equation 7:

$$\text{frame rate} = \frac{2 \times \text{vt\_pixel\_clock\_freq\_mhz} \times 1 \times 10^6}{\text{line\_length\_pck} \times \text{frame\_length\_lines}} \quad \text{(eq. 7)}$$

If coarse_integration_time is set larger than frame_length_lines the frame size will be expanded to coarse_integration_time + 1.

*Minimum Row Time*

The minimum row time and blanking values are shown in Table 14.

**Table 14. MINIMUM ROW TIME AND BLANKING NUMBERS**

| | No Column Binning | Column Binning |
|---|---|---|
| min_line_blanking_pck | 212 | 132 |
| min_line_length_pck | 4640 | 4640 |

In addition, enough time must be given to the output FIFO so it can output all data at the set frequency within one row time. There are therefore three checks that must all be met when programming line_length_pck:

- line_length_pck > min_line_length_pck in Table 14
- line_length_pck > (x_addr_end − x_addr_start + x_odd_inc) / subsampling_factor + min_line_blanking_pck
- line_length_pck > (x_output_size/#_of_mipi_lane + 0x68) × (pix_clock_freq_mhz/word_clock_freq_mhz) × 2

*Minimum Frame Time*

The minimum number of rows in the image is 2, so min_frame_length_lines will always equal (min_frame_ blanking_lines + 2).

**Table 15. MINIMUM FRAME TIME AND BLANKING NUMBERS**

| | No Row Binning | Row Binning |
|---|---|---|
| min_frame_blanking_lines | 0xE | 0xC |

*Integration Time*

The integration (exposure) time of the AR1335 is controlled by the coarse_integration_time registers.

The limits for the coarse integration time are defined by:

$$\text{coarse\_integration\_time\_min} \leq \text{coarse\_integartion\_time} \quad \text{(eq. 8)}$$

The actual integration time is given by:

$$\text{integration\_time} = \frac{\text{coarse\_integration\_time} \times \text{line\_length\_pck}}{2 \times \text{vt\_pixel\_clock\_freq\_mhz} \times 10^6} \quad \text{(eq. 9)}$$

It is required that:

$$\text{coarse\_integration\_time} \leq (\text{frame\_length\_lines} - \text{coarse\_integration\_time\_max\_margin}) \quad \text{(eq. 10)}$$

If this limit is broken, the frame time will automatically be extended to (coarse_integration_time + coarse_integartion_ time_max_margin) to accommodate the larger integration time.

**Flash Timing Control**

The AR1335 supports both xenon and LED flash timing through GPIO[0] (default output). The timing of the FLASH signal with the default settings is shown in Figure 23 (Xenon) and Figure 24 (LED). The flash and flash_count registers allow the timing of the flash to be changed. The flash can be programmed to fire only once, delayed by a few frames when asserted, and (for xenon flash) the flash duration can be programmed.

Enabling the LED flash will cause one bad frame, where several of the rows only have the flash on for part of their integration time. This can be avoided either by first enabling mask bad frames (R0x301A[9] = 1) before the enabling the flash or by forcing a restart (R0x301A[1] = 1) immediately after enabling the flash; the first bad frame will then be masked out, as shown in Figure 24. Read-only bit flash[14] is set during frames that are correctly integrated; the state of this bit is shown in Figures 23 and Figure 24.

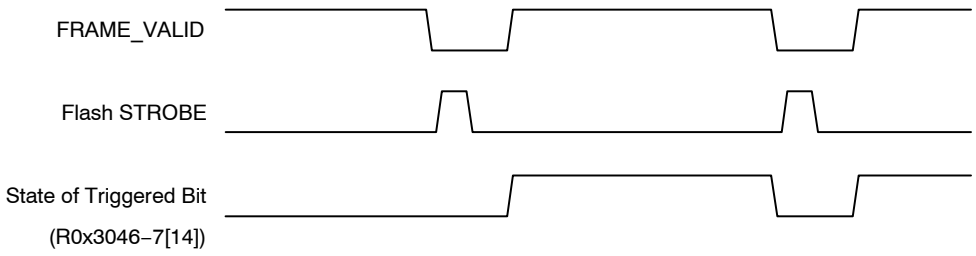
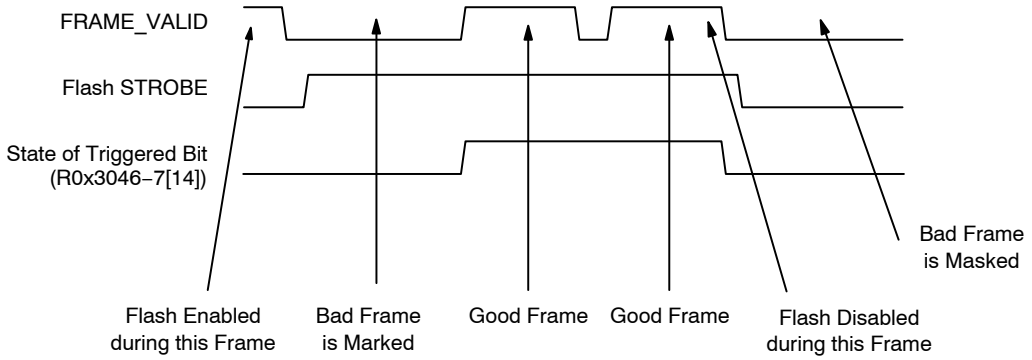



**Figure 23. Xenon Flash Enabled**




NOTE: An option to invert the flash output signal through R0x3046[7] is also available.

**Figure 24. LED Flash Enabled**

## ADDITIONAL FEATURES

### Bit-depth Compression

AR1335 supports SMIA DPCM with 10-8-10 and 10-6-10 compression.

The output compressed data is MSB-aligned. When compression is enabled, ‘0’s are padded to the LSB side.

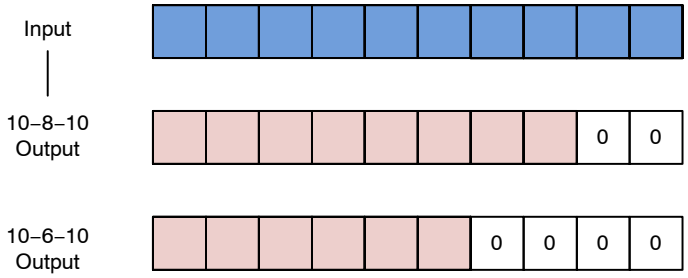



**Figure 25.**

The registers listed in Table 16 are related to compression.

**Table 16. COMPRESSION REGISTERS**

| Address | Name | Register | Description |
|---|---|---|---|
| 0x0112 | data_format | RW | [3:0]: The bit-width of the compressed pixel data<br>[11:8]: The bit-width of the uncompressed data |
| 0x31AE | serial format | RW | [9:8]: Serial interface type<br>2: MIPI<br>3: Reserved<br>[2:0]: serial data lanes |

### 3D Support

This function uses GPI (as an input) and GPIO1 (as an output). The input is sampled on rising CLK (EXTCLK) and the output changes on rising CLK (EXTCLK). In order to use this function, GPIO1 must be enabled as an output.

The CHAIN_CONTROL register should only be written when the sensor is not in streaming mode; the effect of writing to this register when the sensor is in streaming mode is UNDEFINED.

When chain_enable = 1 and master = 0, transitions on the SADDR input are propagated synchronously to the GPIO1 output.

When chain_enable = 1 and master = 1, GPIO1 generates ‘events’ under certain circumstances. An event is a ‘start’ bit followed by a 4−bit code.

When chain_enable = 1 and sync_enable = 1 and master = 0, and the sensor core enters streaming mode (bit 2 in reset_register is changed to 1), the actual entry to streaming mode is delayed until the sensor receives an event.

## MULTI-CAMERA SYNCHRONIZATION

In order to make sure that cameras in a 3D system are working in sync, two synchronization methods are supported, Trigger Mode and Global Start.

### Trigger Mode

The sensors should be wired as Figure 26. A sensor GPI is configured as the trigger pin. Each sensor runs like standalone, except being configured as trigger mode. It needs external pulses to start streaming. The host should transmit the pulses to the sensor trigger pins. All sensors should share the same trigger pulse signal and the same EXTCLK.

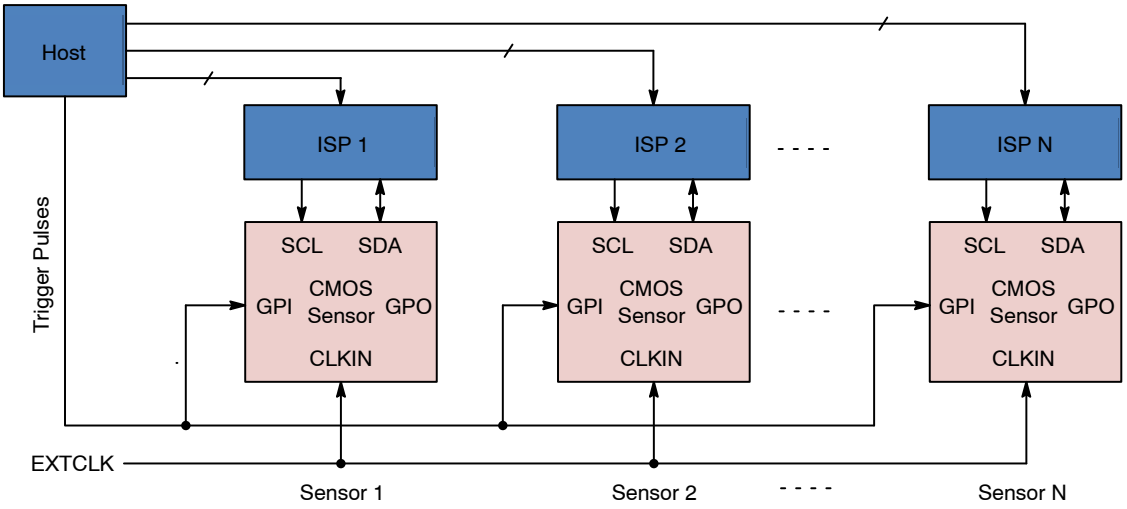



**Figure 26. Trigger Mode Block Diagram**

### Global Start

The sensors should be wired as Figure 27. All sensors in this mode share the same $I^2C$ bus, which is mastered by the hose. A sensor GPI is configured as SADDR pins. The host can broadcast $I^2C$ commands to all sensors or access individual sensor by controlling SADDR pins. To start and synchronize the first frame, the host broadcasts Start Streaming command to all sensors using the common $I^2C$ ID.

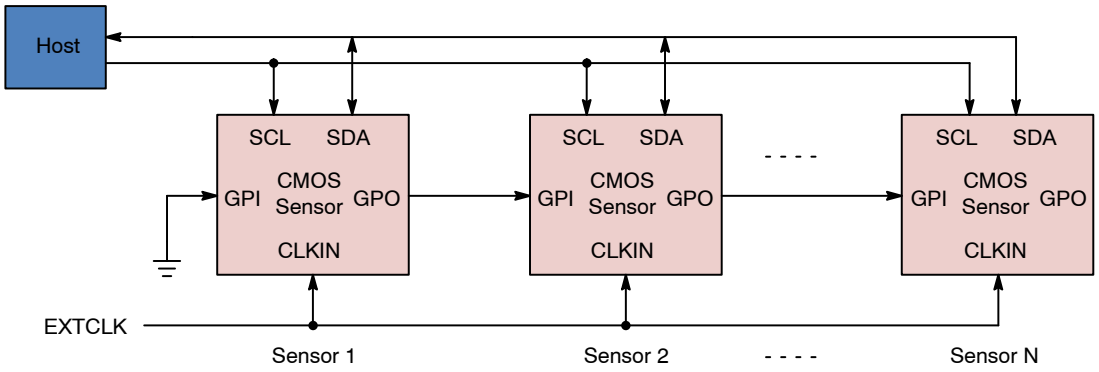



**Figure 27. Daisy-Chained GPIO**

### Integration Time for Interlaced HDR Readout

*Tint1(Integration Time 1) and Tint2 (Integration Time 2)*

The limits for the coarse integration time are defined by:

$$\text{coarse\_integration\_time\_min} \le \text{coarse\_integration\_time} \le (\text{frame\_length\_lines} - \text{coarse\_integration\_time\_max\_margin}) \quad \text{(eq. 11)}$$

$$\text{coarse\_integration\_time2\_min} \le \text{coarse\_integration\_time2} \le (\text{frame\_length\_lines} - \text{coarse\_integration\_time2\_max\_margin}) \quad \text{(eq. 12)}$$

The actual integration time is given by:

$$\text{integration\_time} = \frac{\text{coarse\_integration\_time} \times \text{line\_length\_pck}}{2 \times \text{vt\_pixel\_clock\_freq\_mhz} \times 10^6} \quad \text{(eq. 13)}$$

$$\text{integration\_time2} = \frac{\text{coarse\_integration\_time2} \times \text{line\_length\_pck}}{2 \times \text{vt\_pixel\_clock\_freq\_mhz} \times 10^6} \quad \text{(eq. 14)}$$

If this limit is broken, the frame time will automatically be extended to *(coarse_integration_time + coarse_integration_time_max_margin)* to accommodate the larger integration time.

The ratio between even and odd rows is typically adjusted to 1x, 2x, 4x, and 8x.

## GAIN

AR1335 supports both analog and digital gain.

### Analog Gain

Analog gain is provided by adjusting the column-parallel ADC reference voltage. The fine gain is implemented through adjusting the ADC reference voltage with fine steps. Global gain register (R0x305E) sets the analog gain. Analog coarse gain is set by setting bits R0x305E[6:4] while analog fine gain is set by setting bits R0x305E[3:0]. The maximum analog gain is 7.75x. Table 17 shows the recommended gain settings:

**Table 17. RECOMMENDED GAIN SETTINGS**

| Gain Codes R0x305E[15:0] | Analog Gain | Digital Gain | Total Gain |
|---|---|---|---|
| 0x2010 | 1 | 1 | 1 |
| 0x2014 | 1.25 | 1 | 1.25 |
| 0x2018 | 1.5 | 1 | 1.5 |
| 0x201C | 1.75 | 1 | 1.75 |
| 0x2020 | 2 | 1 | 2 |
| 0x2022 | 2.25 | 1 | 2.25 |
| 0x2024 | 2.5 | 1 | 2.5 |
| 0x2026 | 2.75 | 1 | 2.75 |
| 0x2028 | 3 | 1 | 3 |
| 0x202A | 3.25 | 1 | 3.25 |
| 0x202C | 3.5 | 1 | 3.5 |
| 0x202E | 3.75 | 1 | 3.75 |
| 0x2030 | 4 | 1 | 4 |
| 0x2031 | 4.25 | 1 | 4.25 |
| 0x2032 | 4.5 | 1 | 4.5 |
| 0x2033 | 4.75 | 1 | 4.75 |
| 0x2034 | 5 | 1 | 5 |
| 0x2035 | 5.25 | 1 | 5.25 |
| 0x2036 | 5.5 | 1 | 5.5 |
| 0x2037 | 5.75 | 1 | 5.75 |
| 0x2038 | 6 | 1 | 6 |
| 0x2039 | 6.25 | 1 | 6.25 |
| 0x203A | 6.5 | 1 | 6.5 |
| 0x203B | 6.75 | 1 | 6.75 |
| 0x203C | 7 | 1 | 7 |
| 0x203D | 7.25 | 1 | 7.25 |
| 0x203E | 7.5 | 1 | 7.5 |
| 0x203F | 7.75 | 1 | 7.75 |
| 0x213F | 7.75 | 1.03125 | 8 |
| 0x223F | 7.75 | 1.0625 | 8.25 |
| 0x233F | 7.75 | 1.09375 | 8.5 |
| 0x243F | 7.75 | 1.125 | 8.75 |
| 0x253F | 7.75 | 1.15625 | 9 |
| 0x263F | 7.75 | 1.1875 | 9.25 |
| 0x273F | 7.75 | 1.21875 | 9.5 |

**Table 17. RECOMMENDED GAIN SETTINGS** (continued)

| Gain Codes R0x305E[15:0] | Analog Gain | Digital Gain | Total Gain |
|---|---|---|---|
| 0x28BF | 7.75 | 1.265625 | 9.75 |
| 0x29BF | 7.75 | 1.296875 | 10 |
| 0x2ABF | 7.75 | 1.328125 | 10.25 |
| 0x2BBF | 7.75 | 1.359375 | 10.5 |
| 0x2CBF | 7.75 | 1.390625 | 10.75 |
| 0x2DBF | 7.75 | 1.421875 | 11 |
| 0x2EBF | 7.75 | 1.453125 | 11.25 |
| 0x2FBF | 7.75 | 1.484375 | 11.5 |
| 0x30BF | 7.75 | 1.515625 | 11.75 |
| 0x31BF | 7.75 | 1.546875 | 12 |
| 0x32BF | 7.75 | 1.578125 | 12.25 |
| 0x33BF | 7.75 | 1.609375 | 12.5 |
| 0x34BF | 7.75 | 1.640625 | 12.75 |
| 0x35BF | 7.75 | 1.671875 | 13 |
| 0x36BF | 7.75 | 1.703125 | 13.25 |
| 0x37BF | 7.75 | 1.734375 | 13.5 |
| 0x393F | 7.75 | 1.78125 | 13.75 |
| 0x3A3F | 7.75 | 1.8125 | 14 |
| 0x3B3F | 7.75 | 1.84375 | 14.25 |
| 0x3C3F | 7.75 | 1.875 | 14.5 |
| 0x3D3F | 7.75 | 1.90625 | 14.75 |
| 0x3E3F | 7.75 | 1.9375 | 15 |
| 0x3F3F | 7.75 | 1.96875 | 15.25 |
| 0x403F | 7.75 | 2 | 15.5 |
| 0x413F | 7.75 | 2.03125 | 15.75 |
| 0x423F | 7.75 | 2.0625 | 16 |
| 0x433F | 7.75 | 2.09375 | 16.25 |
| 0x443F | 7.75 | 2.125 | 16.5 |
| 0x453F | 7.75 | 2.15625 | 16.75 |
| 0x463F | 7.75 | 2.1875 | 17 |
| 0x473F | 7.75 | 2.21875 | 17.25 |
| 0x48BF | 7.75 | 2.265625 | 17.5 |
| 0x49BF | 7.75 | 2.296875 | 17.75 |
| 0x4ABF | 7.75 | 2.328125 | 18 |
| 0x4BBF | 7.75 | 2.359375 | 18.25 |
| 0x4CBF | 7.75 | 2.390625 | 18.5 |
| 0x4DBF | 7.75 | 2.421875 | 18.75 |
| 0x4EBF | 7.75 | 2.453125 | 19 |
| 0x4FBF | 7.75 | 2.484375 | 19.25 |
| 0x50BF | 7.75 | 2.515625 | 19.5 |
| 0x51BF | 7.75 | 2.546875 | 19.75 |

**Table 17. RECOMMENDED GAIN SETTINGS** (continued)

| Gain Codes R0x305E[15:0] | Analog Gain | Digital Gain | Total Gain |
|---|---|---|---|
| 0x52BF | 7.75 | 2.578125 | 20 |
| 0x53BF | 7.75 | 2.609375 | 20.25 |
| 0x54BF | 7.75 | 2.640625 | 20.5 |
| 0x55BF | 7.75 | 2.671875 | 20.75 |
| 0x56BF | 7.75 | 2.703125 | 21 |
| 0x57BF | 7.75 | 2.734375 | 21.25 |
| 0x593F | 7.75 | 2.78125 | 21.5 |
| 0x5A3F | 7.75 | 2.8125 | 21.75 |
| 0x5B3F | 7.75 | 2.84375 | 22 |
| 0x5C3F | 7.75 | 2.875 | 22.25 |
| 0x5D3F | 7.75 | 2.90625 | 22.5 |
| 0x5E5F | 7.75 | 2.9375 | 22.75 |
| 0x5F3F | 7.75 | 2.96875 | 23 |
| 0x603F | 7.75 | 3 | 23.25 |
| 0x613F | 7.75 | 3.03125 | 23.5 |
| 0x623F | 7.75 | 3.0625 | 23.75 |
| 0x633F | 7.75 | 3.09375 | 24 |

**Table 18. REFERENCE OF ISO-SPEED-BASED GAIN SETTINGS**

| ISO-speed (Note 1) | Total Gain Value | Register 0x305E Value |
|---|---|---|
| 100 | 1.3125x | 0x2015 |
| 200 | 2.625x | 0x2025 |
| 400 | 5.25x | 0x2035 |
| 800 | 10.53x | 0x2BBF |
| 1,600 | 21.05x | 0x573F |
| 3,200 | 42.1x | 0xAE3F |

1. ISO-speed calculations based on 5100K light source and “AP2” IR-cut filter.

### Digital Gain

Digital gain provides fine (sub 1x) gain. The analog and digital gains are multiplicative to give the total gain. Digital gain is set by setting bits R0x305E[15:7] to set global gain where these 11 bits are designed in 3p6 format, that is, 3 MSB provide gain up to 7x in step of 1x while 6 LSB provide sub-1× gain with a step size of 1/64. This sub-1x gain provides the fine gain control for the sensor.

### Total Gain

Maximum total gain is 7.75x (analog) and 7.98x (digital). The total gain equation can be formulated as:

$$\text{Total gain} = 2^{\text{Analog\_coarse\_gain}-1} \times \frac{16 + \text{analog\_fine\_gain}}{16} \times \frac{\text{digital\_gain}}{64} \quad \text{(eq. 15)}$$

## SENSOR CORE DIGITAL DATA PATH

### Test Patterns

The AR1335 supports a number of test patterns to facilitate system debug. Test patterns are enabled using test_pattern_mode (R0x0600–1). The test patterns are listed in Table 19.

**Table 19. TEST PATTERNS**

| test_pattern_mode | Description |
|---|---|
| 0 | Normal Operation: No Test Pattern |
| 1 | Solid Color |
| 2 | 100% Color Bars |
| 3 | Fade-to-Gray Color Bars |
| 256 | Walking 1s (10-bit) |
| 257 | Walking 1s (8-bit) |

Test pattern 0−3 replaces pixel data in the output image (the embedded data rows are still present). Test pattern 4 replaces all data in the output image (the embedded data rows are omitted and test pattern data replaces the pixel data).

For all of the test patterns, the AR1335 registers must be set appropriately to control the frame rate and output timing. This includes:

- All clock divisors
- x_addr_start
- x_addr_end
- y_addr_start
- y_addr_end
- frame_length_lines
- line_length_pck
- x_output_size
- y_output_size

*Effect of Data Path Processing on Test Patterns*

Test patterns are introduced early in the pixel data path. As a result, they can be affected by pixel processing that occurs within the data path. This includes:

- Black pedestal adjustment
- Lens and color shading correction

These effects can be eliminated by the following register settings:

- R0x3044–5[10] = 0
- R0x30CA–B[0] = 1
- R0x30D4–5[15] = 0
- R0x31E0–1[0] = 0
- R0x301A–B[3] = 0 (enable writes to data pedestal)
- R0x301E–F = 0x0000 (set data pedestal to “0”)
- R0x3780[15] = 0 (turn off lens/shading correction)

*Solid Color Test Pattern*

In this mode, all pixel data is replaced by fixed Bayer pattern test data. The intensity of each pixel is set by its associated test data register.

*100% Color Bars Test Pattern*

In this test pattern, shown in Figure 28, all pixel data is replaced by a Bayer version of an 8-color, color-bar chart (white, yellow, cyan, green, magenta, red, blue, black). Each bar is 1/8 of the width of the pixel array (4208/8 = 526 pixels). The pattern repeats after $8 \times 526 = 4208$ pixels. Each color component of each bar is set to either 0 (fully off) or 0x3FF (fully on for 10-bit data).

The pattern occupies the full height of the output image. The image size is set by x_addr_start, x_addr_end, y_addr_start, y_addr_end and may be affected by the setting of x_output_size, y_output_size. The color-bar pattern is disconnected from the addressing of the pixel array, and will therefore always start on the first visible pixel, regardless of the value of x_addr_start. The number of colors that are visible in the output is dependent upon x_addr_end − x_addr_start and the setting of x_output_size: the width of each color bar is fixed at 526 pixels.

The effect of setting horizontal_mirror in conjunction with this test pattern is that the order in which the colors are generated is reversed: the black bar appears at the left side of the output image. Any pattern repeat occurs at the right side of the output image regardless of the setting of horizontal_mirror. The state of vertical_flip has no effect on this test pattern.

The effect of subsampling and scaling of this test pattern is undefined.

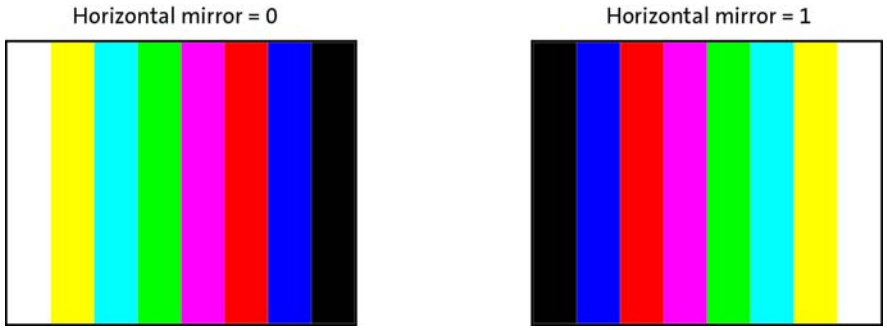



**Figure 28. 100% Color Bars Test Pattern**

*Fade-to-Gray Color Bars Test Pattern*

In this test pattern, shown in Figure 29, all pixel data is replaced by a Bayer version of an 8-color, color-bar chart (white, yellow, cyan, green, magenta, red, blue, black). Each bar is 1/8 of the width of the pixel array (4208/8 = 526 pixels). The test pattern repeats after 4208 pixels.

Each color bar fades vertically from zero or full intensity at the top of the image to 50% intensity (mid-gray). The pattern of periodic repeating rows is corresponding to half the full intensity. For example, repeating every 512 rows in 10-bit mode. Each color bar is divided into a left and a right half, in which the left half fades smoothly and the right half fades in quantized steps.

The speed at which each color fades is dependent on the sensor's data width and the height of the pixel array. We want half of the data range (from 100 or 0 to 50 percent) difference between the top and bottom of the pattern. Because of the Bayer pattern, each state must be held for two rows.

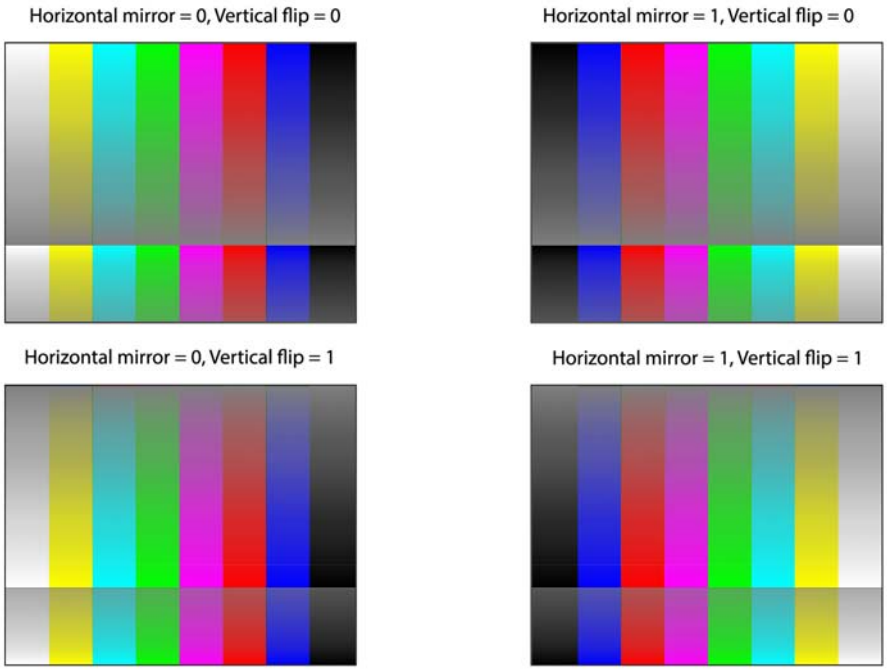



**Figure 29. Fade-to-Gray Color Bars Test Pattern**

The rate-of-fade of the Bayer pattern is set so that there is at least one full pattern within a full-sized image for the sensor. Factors that affect this are the resolution of the ADC (10-bit) and the image height. For example, the AR1335 fades the pixels by 2 LSB for each two rows. With 10-bit data, the pattern is 3120 pixels high and repeats after that, if the window is higher.

The image size is set by x_addr_start, x_addr_end, y_addr_start, y_addr_end and may be affected by the setting of x_output_size, y_output_size. The color-bar pattern starts at the first column in the image, regardless of the value of x_addr_start. The number of colors that are visible in the output is dependent upon x_addr_end – x_addr_start and the setting of x_output_size: the width of each color bar is fixed at 526 pixels.

The effect of setting horizontal_mirror or vertical_flip in conjunction with this test pattern is that the order in which the colors are generated is reversed: the black bar appears at

the left side of the output image. Any pattern repeat occurs at the right side of the output image regardless of the setting of horizontal_mirror.

The effect of subsampling and scaling of this test pattern is undefined.

### Walking 1s

When selected, a walking 1s pattern will be sent through the digital pipeline. The first value in each row is 0. Each value will be valid for 2 pixels.

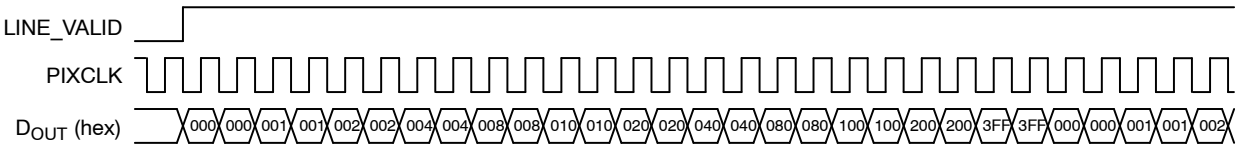



**Figure 30. Walking 1s 10-bit Pattern**

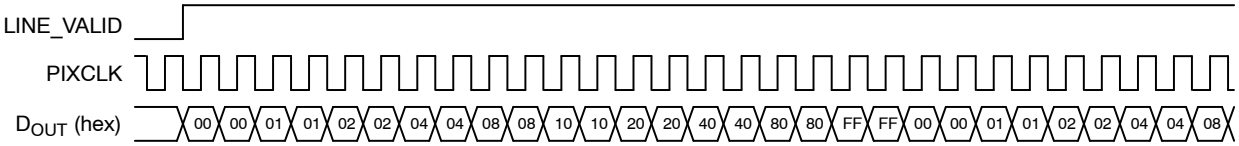



**Figure 31. Walking 1s 8-bit Pattern**

The walking 1s pattern was implemented to facilitate assembly testing of modules.

The walking 1 test pattern is not active during the blanking periods, hence the output would reset to a value of 0x0. When the active period starts again, the pattern would restart from the beginning. The behavior of this test pattern is the same between full resolution and subsampling modes. RAW10 and RAW8 walking 1 modes are enabled by different test pattern codes.

### Pedestal

This block adds the value from R0x0008–9 or (data_pedestal_) to the incoming pixel value.

The data_pedestal register is read-only by default but can be made read/write by clearing the lock_reg bit in R0x301A–B.

The only way to disable the effect of the pedestal is to set it to 0.

## TIMING SPECIFICATIONS

By design, the 1.8 V $DV_{DD}$_IO is an always-on-power domain. The 1.2 V $DV_{DD}$_PHY is used in MIPI and is also always-on without power switching. The rest of the digital power supplies ($DV_{DD}$, $DV_{DD}$_PLL) and the analog power supplies ($V_{AA}$, $V_{AA}$PIX, $V_{DD}$IO_ANA, and $DV_{DD}$_ANA) are switched power supplies. These power supplies are disconnected from the chip core by power switches in hard standby mode.

### Power-Up Sequence

The recommended power-up sequence for the AR1335 is shown in Figure 32.

1. Set XSHUTDOWN LOW.
2. Power up VDD_IO (1.8 V only), VDDIO_ANA (1.8 V only).
3. After 1−500 ms, power up VDD, VDD_ANA, VDD_PLL, VDD_PHY (1.2 V) and VAA, VAA_PIX (2.7 V) (any order).
4. Apply EXTCLK (can be applied anytime).
5. After 1−500 ms, set XSHUTDOWN HIGH.
6. HOST configuration through $I^2C$ (PLL, output, etc).
7. Set mode_select = 1 bit.
8. PLL internally enables and locks.
9. AR1335 enters streaming mode.

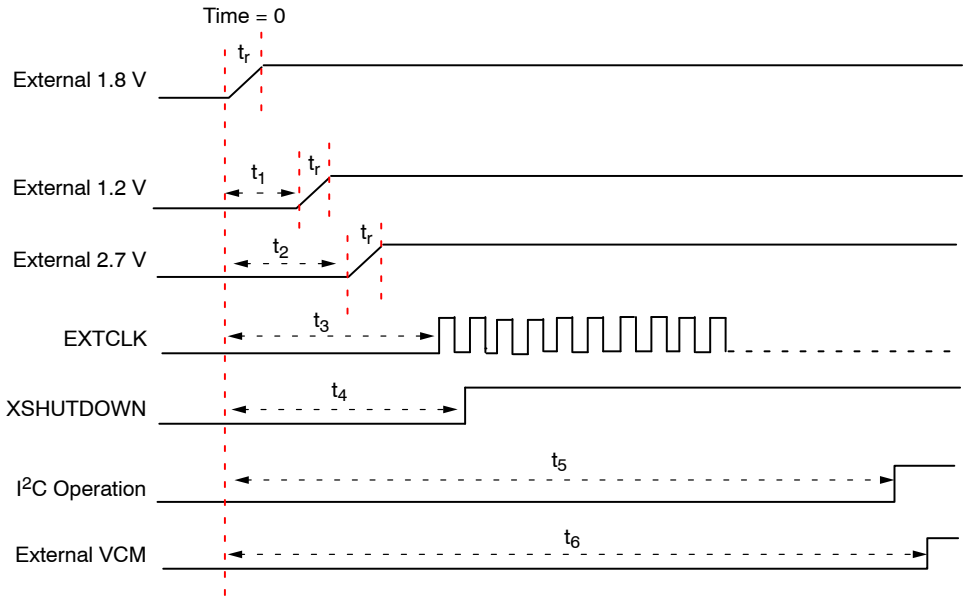



**Figure 32. Power-Up Sequence**

**Table 20. POWER-UP SEQUENCE**

| Timing | Minimum | Maximum |
|---|---|---|
| $t_1$ | 1 ms | – |
| $t_2$ | $t_1$ + 1 ms | – |
| $t_3$ | – | < $t_4$ |
| $t_4$ | $t_2$ + 1 ms | – |
| $t_5$ | $t_4$ + 1 ms | – |
| $t_r$ | 100 μs | – |

A minimum ramping time of 100 μS for "$t_r$" is needed to minimize surge of supply current when powering up the 1.2 V, 1.8 V, and 2.7 V external power supplies.

### Power-Down Sequence

The recommended power-down sequence for the AR1335 is shown in Figure 33. The available power supplies – $V_{DD}$_IO, $V_{DD}$IO_ANA, $V_{DD}$, $V_{DD}$_ANA, $V_{DD}$_PLL, $V_{DD}$_PHY, $V_{AA}$, $V_{AA}$_PIX – can be turned off at the same time or have the separation specified below.

1. Set software standby mode (mode_select = 0) register.
2. Wait till the end of the current frame.
3. Set XSHUTDOWN LOW.
4. Power off $V_{DD}$, $V_{DD}$_PLL, $V_{DD}$_PHY, (1.2 V) and $V_{AA}$, $V_{AA}$_PIX (2.7 V) (any order).
5. Power off $V_{DD}$_IO (1.8 V).

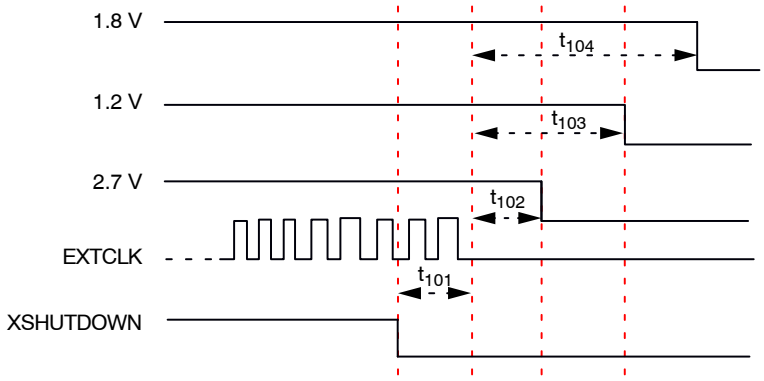



**Figure 33. Power-Down Sequence**

**Table 21. RECOMMENDED POWER-DOWN TIMING**

| Timing | Minimum | Maximum |
|---|---|---|
| $t_{101}$ | 10 μs | 10 ms |
| $t_{102}$ | 10 ms | – |
| $t_{103}$ | 10 ms | – |
| $t_{104}$ | max($t_{102}$, $t_{103}$) + 10 ms | – |

### Hard Standby and Hard Reset

The hard standby state is reached by the assertion of the XSHUTDOWN pads. Register values are not retained by this action, and will be returned to their default values once hard reset is completed. The minimum power consumption is achieved by the hard standby state. The details of the sequence are described below and shown in Figure 34.

1. Disable streaming if output is active by setting mode_select = 0 (R0x0100).
2. The soft standby state is reached after the current row or frame, depending on configuration, has ended.
3. Assert XSHUTDOWN (active LOW) and de-assert XSHUTDOWN to reset the sensor.
4. The sensor remains in hard standby state if XSHUTDOWN remain in the logic “0” state.

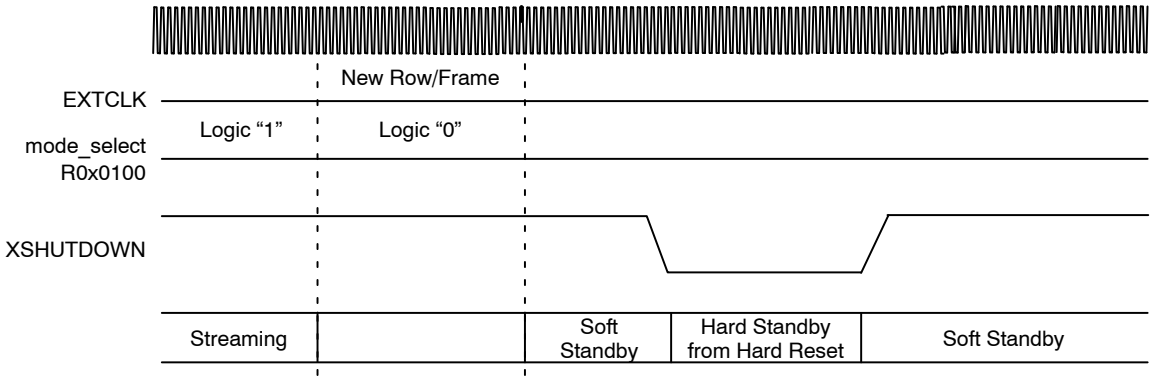



**Figure 34. Hard Standby and Hard Reset**

### Soft Standby and Soft Reset

The AR1335 can reduce power consumption by switching to the soft standby state when the output is not needed. Register values are retained in the soft standby state. Once this state is reached, soft reset can be enabled optionally to return all register values back to the default. The details of the sequence are described below and shown in Figure 35.

*Soft Standby*

1. Disable streaming if output is active by setting mode_select = 0 (R0x0100).
2. The soft standby state is reached after the current row or frame, depending on configuration, has ended.
3. The soft standby with external clock disabled retains register values, uses minimum power, and allows faster power-up.

*Soft Reset*

1. Follow the soft standby sequence list above.
2. Set software_reset = 1 (R0x0103) to start the internal initialization sequence.
3. After 15000 EXTCLKs, the internal initialization sequence is completed and the current state returns to soft standby automatically. All registers, including software_reset, returns to their default values.

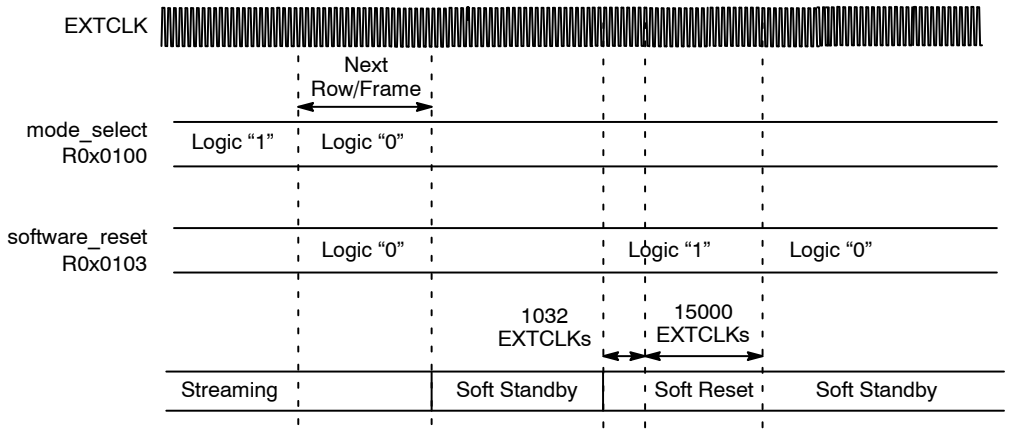



**Figure 35. Soft Standby and Soft Reset**

## IMAGE SENSOR CHARACTERISTICS

**Table 22. IMAGE SENSOR CHARACTERISTICS**

(All specifications address operation at $T_J$ = 60°C (±3°C), $V_{AA}$ = $V_{AA}$_PIX = 2.7 V, $V_{DD}$ = $V_{DD}$_ANA = $V_{DD}$_PHY = 1.2 V, $V_{DD}$_IO = $V_{DD}$IO_ANA = 1.8 V. Unless otherwise specified, all tests are performed with ON Semiconductor-recommended settings in full resolution, max FPS mode at minimum recommended gain and resulting images are averaged and analyzed.)

| Item | Symbol | Min | Typ | Max | Unit | Region | Measurement Method | Comments |
|---|---|---|---|---|---|---|---|---|
| Responsivity | S | | 4700 | | $e^-$/lx.sec | Center | 1.0 | |
| Responsivity Ratio | RG | 0.54 | 0.61 | 0.68 | | Center | 1.1 | 1x (Minimum Gain) |
| | BG | 0.39 | 0.44 | 0.49 | | | | |

*Measurement Method*

- *1.0* – QE spectrum vs wavelength is measured using a monochromator. Responsivity is calculated from the green QE curve using a 5100K incident light spectrum filtered by a UV640 IR-cut filter. The incident light spectrum is then converted to lux by using the photopic conversion spectrum. Then the number of signal electrons generated at each wavelength, and the incident lux on the sensor plane at each wavelength are both integrated, and then divided, resulting in the responsivity value of electrons per lux*s.
- *1.1* – Use a 3000K uniform Halogen illuminator with a CM500 IR-cut filter and a lens/aperture to be within 3° of the designed CRA vs. image height. Then set the light level to achieve a Ga signal response of approximately 512 LSB, where Ga is the average Gr and Gb signal response. Read out the B, Gr, Gb, and R signal outputs from the center 200 × 200 window after subtracting the mean black-level correction offsets and calculate ratios as per below:

$$Ga = \frac{Gr + Gb}{2}, \text{ and} \quad \text{(eq. 16)}$$

$$BG = \frac{B}{Ga}, \text{ and} \quad \text{(eq. 17)}$$

$$RG = \frac{R}{Ga} \quad \text{(eq. 18)}$$

### Spectral Characteristics

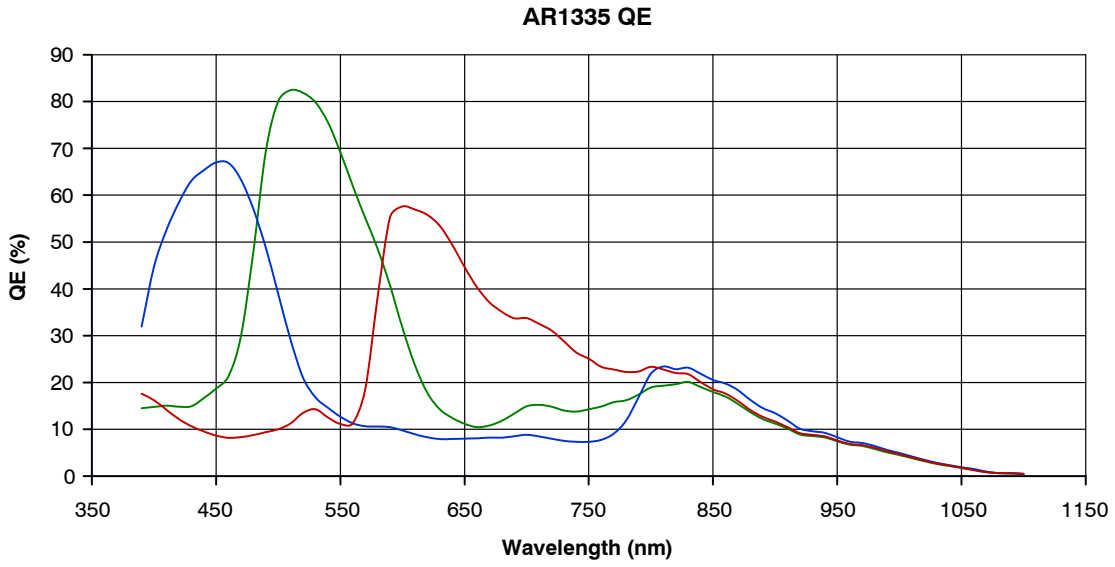



**Figure 36. AR1335 Quantum Efficiency vs. Wavelength**

### Chief Ray Angle vs. Image Height

CRA vs. Image Height Plot

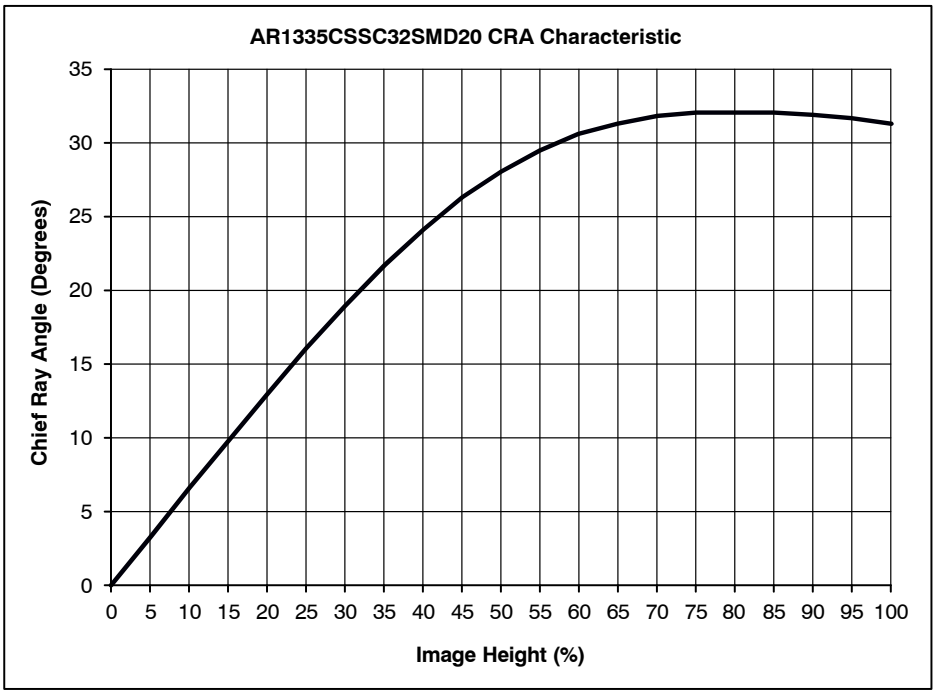



**Figure 37. AR1335CSSC32SMD20 Chief Ray Angle vs. Image Height**

| Image Height | | CRA |
|---|---|---|
| (%) | (mm) | (deg) |
| 0 | 0 | 0 |
| 5 | 0.144 | 3.36 |
| 10 | 0.288 | 6.62 |
| 15 | 0.432 | 9.82 |
| 20 | 0.576 | 12.98 |
| 25 | 0.720 | 16.04 |
| 30 | 0.864 | 18.97 |
| 35 | 1.008 | 21.69 |
| 40 | 1.152 | 24.14 |
| 45 | 1.297 | 26.28 |
| 50 | 1.441 | 28.07 |
| 55 | 1.585 | 29.50 |
| 60 | 1.729 | 30.57 |
| 65 | 1.873 | 31.32 |
| 70 | 2.017 | 31.78 |
| 75 | 2.161 | 32.01 |
| 80 | 2.305 | 32.07 |
| 85 | 2.449 | 32.01 |
| 90 | 2.593 | 31.87 |
| 95 | 2.737 | 31.65 |
| 100 | 2.881 | 31.31 |

**Table 23. AR1335CSSC32SMD20 ARRAY DIMENSIONS**

| Pixel Size | Array Size (Pixels) | | Array Size (mm) | |
|---|---|---|---|---|
| 1.1 μm | $X_{Max}$ | 4208 | $X_{Max}$ | 4.63 |
| | $Y_{Max}$ | 3120 | $Y_{Max}$ | 3.43 |
| | | | Diagonal | 5.76 |

*Read the Sensor CRA*

Follow the steps below to obtain the CRA value of the image sensor:

1. Set the register bit field R0x301A[5] = 1.
2. Read the register bit fields R0x31FE [9:7].
3. Determine the CRA value according to Table 24.

**Table 24. AR1335CSSC32SMD20 CRA VALUE**

| Binary Value of R0x31FE[9:7] | CRA Value |
|---|---|
| 001 | 32 |

CRA vs. Image Height Plot

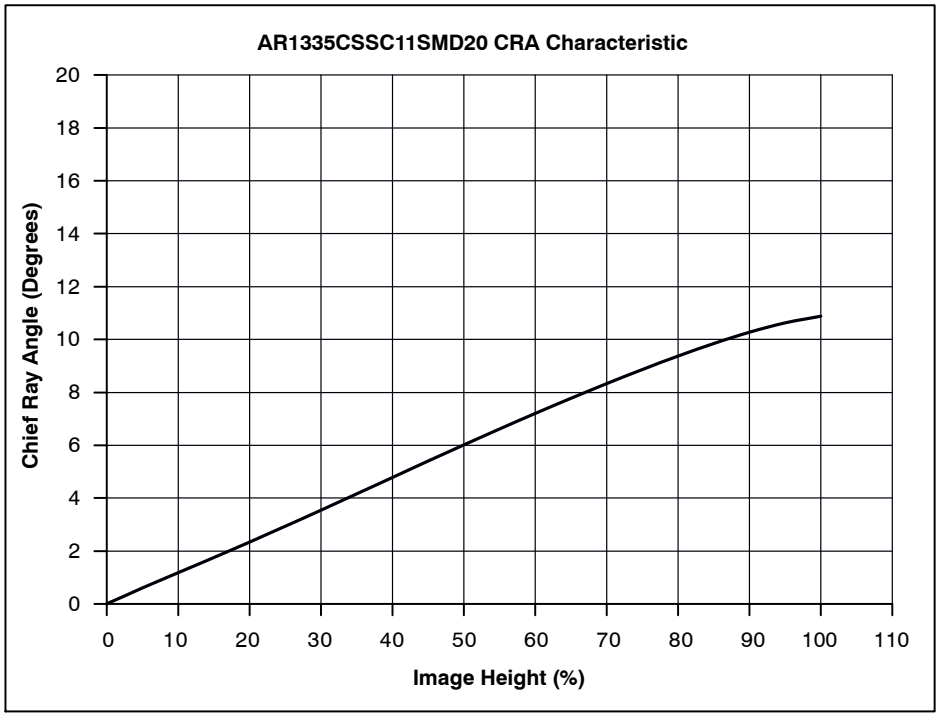



**Figure 38. AR1335CSSC11SMD20 Chief Ray Angle vs. Image Height**

| Image Height | | CRA |
|---|---|---|
| (%) | (mm) | (deg) |
| 0 | 0 | 0 |
| 5 | 0.144 | 0.60 |
| 10 | 0.288 | 1.18 |
| 15 | 0.432 | 1.75 |
| 20 | 0.576 | 2.34 |
| 25 | 0.720 | 2.93 |
| 30 | 0.864 | 3.54 |
| 35 | 1.008 | 4.16 |
| 40 | 1.152 | 4.78 |
| 45 | 1.297 | 5.40 |
| 50 | 1.441 | 6.01 |
| 55 | 1.585 | 6.61 |
| 60 | 1.729 | 7.20 |
| 65 | 1.873 | 7.77 |
| 70 | 2.017 | 8.33 |
| 75 | 2.161 | 8.86 |
| 80 | 2.305 | 9.37 |
| 85 | 2.449 | 9.85 |
| 90 | 2.593 | 10.27 |
| 95 | 2.737 | 10.63 |
| 100 | 2.881 | 10.88 |

**Table 25. AR1335CSSC11SMD20 ARRAY DIMENSIONS**

| Pixel Size | Array Size (Pixels) | | Array Size (mm) | |
|---|---|---|---|---|
| 1.1 μm | $X_{Max}$ | 4208 | $X_{Max}$ | 4.63 |
| | $Y_{Max}$ | 3120 | $Y_{Max}$ | 3.43 |
| | | | Diagonal | 5.76 |

*Read the Sensor CRA*

Follow the steps below to obtain the CRA value of the image sensor:

1. Set the register bit field R0x301A[5] = 1.
2. Read the register bit fields R0x31FE [9:7].
3. Determine the CRA value according to Table 26.

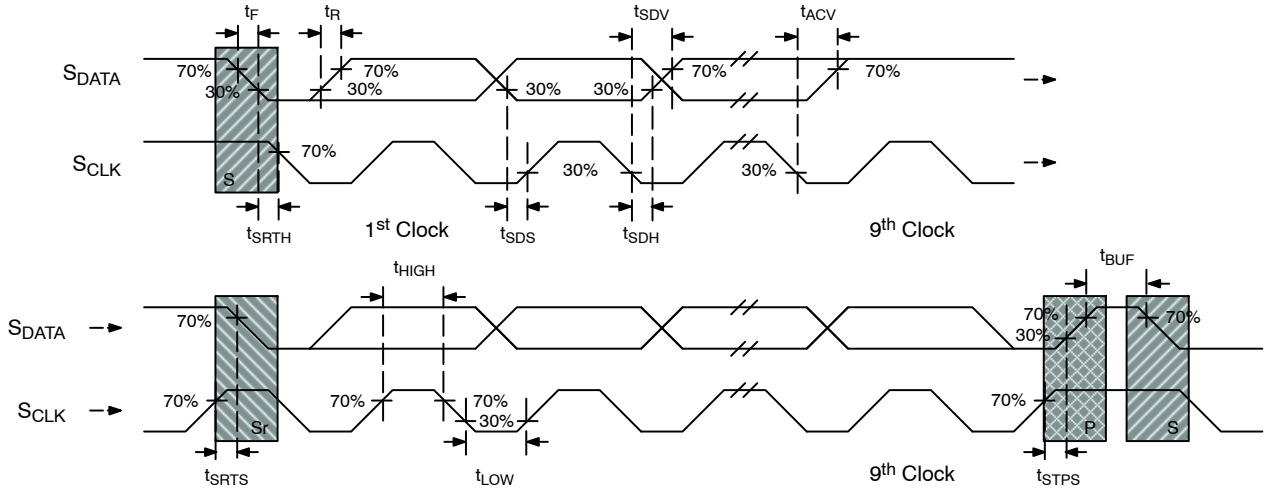
**Table 26. AR1335CSSC11SMD20 CRA VALUE**

| Binary Value of R0x31FE[9:7] | CRA Value |
|---|---|
| 0x2 | 11 |

## ELECTRICAL CHARACTERISTICS

### Two-Wire Serial Register Interface

The electrical characteristics of the two-wire serial register interface ($S_{CLK}$, $S_{DATA}$) are shown in Figure 39 and Table 27. Table 28 shows the timing specification for the two-wire serial interface.




NOTE: Read sequence: For an 8-bit READ, read waveforms start after WRITE command and register address are issued.

**Figure 39. Two-Wire Serial Bus Timing Parameters**

**Table 27. TWO-WIRE SERIAL REGISTER INTERFACE ELECTRICAL CHARACTERISTICS**

($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA\_PIX}$ = 2.7 V; $V_{DD\_IO}$ = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD\_PHY}$ = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Fast Mode | | | Fast Mode Plus | | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min. | Typ. | Max. | Min. | Typ. | Max | |
| **$S_{DATA}$** | | | | | | | | | |
| $V_{IL}$ | Input LOW Voltage | – | –0.3 | – | 0x3 × $V_{DD\_IO}$ | –0.3 | – | 0x3 × $V_{DD\_IO}$ | V |
| $V_{IH}$ | Input HIGH Voltage | – | 0.7 × $V_{DD\_IO}$ | – | $V_{DD\_IO}$ + 0.3 | 0.7 × $V_{DD\_IO}$ | – | $V_{DD\_IO}$ + 0.3 | V |
| $I_{IL}$ | Input Leakage Current | No pull up resistor; $V_{IN}$ = $V_{DD\_IO}$ or $D_{GND}$ | 10 | – | 14 | 10 | – | 14 | μA |
| $V_{OL}$ | Output LOW Voltage | At specified 2 mA | 0.11 | – | 0.3 | 0.11 | – | 0.3 | V |
| $I_{OL}$ | Output LOW Current | At specified $V_{OL}$ 0.4 V | 3 | – | N/A | 20 | – | N/A | mA |
| $C_{IN}$ | Input Pad Capacitance | – | 2.4 | – | 2.62 | 2.4 | – | 2.62 | pF |
| $C_{LOAD}$ | Load Capacitance | – | – | – | N/A | – | – | N/A | pF |

**Table 27. TWO-WIRE SERIAL REGISTER INTERFACE ELECTRICAL CHARACTERISTICS** (continued)
($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Fast Mode | | | Fast Mode Plus | | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min. | Typ. | Max. | Min. | Typ. | Max | |
| **$S_{CLK}$** | | | | | | | | | |
| $V_{IL}$ | Input LOW Voltage | – | –0.3 | – | 0x3 × $V_{DD}$_IO | – | – | – | V |
| $V_{IH}$ | Input HIGH Voltage | – | 0.7 × $V_{DD}$_IO | – | $V_{DD}$_IO + 0.3 | – | – | – | V |
| $I_{IL}$ | Input Leakage Current | No pull up resistor; $V_{IN}$ = $V_{DD}$_IO or $D_{GND}$ | 10 | – | 14 | – | – | – | μA |
| $V_{OL}$ | Output LOW Voltage | At specified 2 mA | 0.11 | – | 0.3 | – | – | – | V |
| $I_{OL}$ | Output LOW Current | At specified $V_{OL}$ 0.1 V | 3 | – | N/A | – | – | – | mA |
| $C_{INL}$ | Input Pad Capacitance | – | 2.4 | – | 2.63 | – | – | – | pF |
| $C_{LOAD}$ | Load Capacitance | – | – | – | N/A | – | – | – | pF |

Product parametric performance is indicated in the Electrical Characteristics for the listed test conditions, unless otherwise noted. Product performance may not be indicated by the Electrical Characteristics if operated under different conditions.

**Table 28. TWO-WIRE SERIAL TIMING SPECIFICATIONS**

| Symbol | Definition | Fast Mode | | Fast Mode Plus | | Unit |
|---|---|---|---|---|---|---|
| | | Min. | Max. | Min. | Max | |
| $f_{SCLK}$ | $S_{CLK}$ Frequency | 0 | 400 | 0 | 1000 | kHz |
| $t_{HIGH}$ | $S_{CLK}$ High Period | 0.6 | – | 0.26 | – | μs |
| $t_{LOW}$ | $S_{CLK}$ Low Period | 1.3 | – | 0.5 | – | μs |
| $t_{SRTS}$ | START Setup Time | 0.6 | – | 0.26 | – | μs |
| $t_{SRTH}$ | START Hold Time | 0.6 | – | 0.26 | – | μs |
| $t_{SDS}$ | Data Setup Time | 100 | – | 50 | – | ns |
| $t_{SDH}$ | Data Hold Time | 0 | – | 0 | – | μs |
| $t_{SDV}$ | Data Valid Time | – | 0.9 | – | 0.45 | μs |
| $t_{ADV}$ | Data Valid Acknowledge Time | – | 0.9 | – | 0.45 | μs |
| $t_{STPS}$ | Stop Setup Time | 0.6 | – | 0.26 | – | μs |
| $t_{BUF}$ | Bus Free Time between STOP and START | 1.3 | – | 0.5 | – | μs |
| $t_R$ | $S_{CLK}$ and $S_{DATA}$ Rise Time | 20 | 300 | – | 120 | ns |
| $t_F$ | $S_{CLK}$ and $S_{DATA}$ Fall Time | 20 | 300 | 20 | 120 | ns |

### EXTCLK

The electrical characteristics of the EXTCLK input are shown in Table 29. The EXTCLK input supports an AC-coupled sine-wave input clock or a DC-coupled square-wave input clock.

If EXTCLK is AC-coupled to the AR1335 and the clock is stopped, the EXTCLK input to the AR1335 must be driven to ground or to $V_{DD}$_IO. Failure to do this will result in excessive current consumption within the EXTCLK input receiver.

**Table 29. ELECTRICAL CHARACTERISTICS (EXTCLK)**

($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $f_{SCLK}$ | $S_{CLK}$ Frequency | 0 | 400 | 0 | 1000 | kHz |
| $f_{EXTCLK}$ | Input Clock Frequency | PLL Enabled | 6 | – | 48 | MHz |
| $t_R$ | Input Clock Rise Slew Rate | $C_{LOAD}$ < 20 pF | – | 2.883 | – | ns |
| $t_F$ | Input Clock Fall Slew Rate | $C_{LOAD}$ < 20 pF | – | 2.687 | – | ns |
| $V_{IN\_AC}$ | Input Clock Minimum Voltage Swing (AC Coupled) | – | 0.5 | – | – | V (p–p) |
| $V_{IN\_DC}$ | Input Clock Maximum Voltage Swing (DC Coupled) | – | – | – | $V_{DD}$_IO + 0.5 | V |
| $f_{CLKMAX(AC)}$ | Input Clock Signaling Frequency (Low Amplitude) | $V_{IN}$ = $V_{IN\_AC}$ (MIN) | – | – | 25 | MHz |
| $f_{CLKMAX(DC)}$ | Input Clock Signaling Frequency (Full Amplitude) | $V_{IN}$ = $V_{DD}$_IO | – | – | 48 | MHz |
| | Clock Duty Cycle | – | 45 | 50 | 55 | % |
| $t_{JITTER}$ | Input Clock Jitter | Cycle-to-Cycle | – | 545 | 600 | ps |
| $t_{LOCK}$ | PLL VCO Lock Time | – | – | 0.2 | 1 | ms |
| $C_{IN}$ | Input Pad Capacitance | – | – | 6 | – | pF |
| $I_{IH}$ | Input HIGH Leakage Current | – | 0 | – | 10 | μA |
| $V_{IH}$ | Input HIGH Voltage | – | 0.7 × $V_{DD}$_IO | | $V_{DD}$_IO + 0.5 | V |
| $V_{IL}$ | Input LOW Voltage | – | –0.5 | | 0.3 × $V_{DD}$_IO | V |

Product parametric performance is indicated in the Electrical Characteristics for the listed test conditions, unless otherwise noted. Product performance may not be indicated by the Electrical Characteristics if operated under different conditions.

### Serial Pixel Data Interface

The electrical characteristics of the serial pixel data interface are shown in Table 30.

To operate the serial pixel data interface within the electrical limits of the CSI−2 and MIPI specifications, $V_{DD}$_IO (I/O digital voltage) is restricted to operate in the range 1.7–1.9 V.

**Table 30. ELECTRICAL CHARACTERISTICS (SERIAL MIPI PIXEL DATA INTERFACE)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $V_{OD}$ | High Speed Transmit Differential Voltage | 140 | – | 270 | mV |
| $V_{CMTX}$ | High Speed Transmit Static Common-mode Voltage | 150 | – | 250 | mV |
| $\Delta V_{CMTX}$ | $V_{CMTX}$ Mismatch when Output is Differential-1 or Differential-0 | < 5 | – | – | mV |
| $\Delta V_{OD}$ | VOD Mismatch when Output is Differential-1 or Differential-0 | < 10 | – | – | mV |
| $Z_{OS}$ | Single Ended Output Impedance | 40 | – | 62.5 | Ω |
| $\Delta Z_{OS}$ | Single Ended Output Impedance Mismatch | < 10 | – | – | % |
| $\Delta V_{CMTX(L,F)}$ | Common-level Variation between 50−450 MHz | < 25 | – | – | mV |
| $t_R$ | Rise Time (20−80%) | 150 | – | 0.3 × UI | ps |
| $t_F$ | Fall Time (20−80%) | 150 | – | 0.3 × UI | ps |
| $V_{OL}$ | Output LOW Level | −50 | – | 50 | mV |
| $V_{OH}$ | Output HIGH Level | 1.1 | – | 1.3 | V |
| $Z_{OLP}$ | Output Impedance of Low Power Parameter | 110 | – | – | Ω |
| $T_{RLP}$ | 15−85% Rise Time | – | – | 25 | ns |
| $T_{FLP}$ | 15−85% Fall Time | – | – | 25 | ns |
| ΔV/Δdtsr | Slew Rate ($C_{LOAD}$ = 0−70 pF) | 30 | – | – | mV/ns |

Product parametric performance is indicated in the Electrical Characteristics for the listed test conditions, unless otherwise noted. Product performance may not be indicated by the Electrical Characteristics if operated under different conditions.

### MIPI Specification Reference

The sensor design and this documentation is based on the following MIPI specifications:

- MIPI Alliance Standard for CSI−2 version 1.0
- MIPI Alliance Standard for D−PHY version 1.0

### Control Interface

The electrical characteristics of the control interface (XSHUTDOWN, TEST, GPI0, GPI1, GPI2, and GPI3) are shown in Table 31.

**Table 31. DC ELECTRICAL CHARACTERISTICS (CONTROL INTERFACE)**

($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input HIGH Voltage | | 0.7 × $V_{DD}$_IO | – | $V_{DD}$_IO + 0.5 | V |
| $V_{IL}$ | Input LOW Voltage | | –0.5 | – | 0.3 × $V_{DD}$_IO | V |
| $I_{IN}$ | Input Leakage Current | No Pull-up Resistor; $V_{IN}$ = $V_{DD}$_IO or DGND | – | – | 10 | μA |
| $C_{IN}$ | Input Pad Capacitance | | – | 6 | – | pF |

### Operating Voltages

$V_{AA}$ and $V_{AA}$_PIX must be at the same potential for correct operation of the AR1335.

**Table 32. DC ELECTRICAL DEFINITIONS AND CHARACTERISTICS**

($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{DD}$_IO | I/O Digital Voltage | – | 1.7 | 1.8 | 1.9 | V |
| $V_{DD}$ | Digital Voltage | – | 1.14 | 1.2 | 1.3 | V |
| $V_{DD}$_PHY | PHY Digital Voltage | – | 1.14 | 1.2 | 1.3 | V |
| $V_{AA}$ | Analog Voltage | – | 2.6 | 2.7 | 2.9 | V |
| $V_{AA}$_PIX | Pixel Supply Voltage | – | 2.6 | 2.7 | 2.9 | V |
| **OPERATING CURRENT** | | | | | | |
| $I_{DD}$_IO | I/O Digital Current | 13M full mode (4208 × 3120), 30 fps | – | 26 | 26 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 118 | 130 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 30 | 30 | |
| $I_{DD}$_IO | I/O Digital Current | 13M full mode (4208 × 3120), 24 fps | – | 26 | 26 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 102 | 113 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 30 | 30 | |
| $I_{DD}$_IO | I/O Digital Current | 4K UHD full mode (3840 × 2160), 30 fps | – | 26 | 26 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 90 | 100 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 28 | 28 | |
| $I_{DD}$_IO | I/O Digital Current | 4K Cinema full mode (4096 × 2160), 30 fps | – | 26 | 26 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 93 | 103 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 28 | 28 | |

**Table 32. DC ELECTRICAL DEFINITIONS AND CHARACTERISTICS** (continued)
($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **OPERATING CURRENT** | | | | | | |
| $I_{DD}$_IO | I/O Digital Current | 1080p (1920 × 1080), crop + bin2 + scaling, 30 fps | – | 8 | 8 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 86 | 94 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 16 | 16 | |
| $I_{DD}$_IO | I/O Digital Current | 1080p LP (1920 × 1080), crop + bin2 + scaling, 30 fps | – | 8 | 8 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 65 | 71 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 16 | 16 | |
| $I_{DD}$_IO | I/O Digital Current | 720p (1280 × 720), crop + skip2 + scaling, 30 fps | – | 6 | 6 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 75 | 83 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 15 | 15 | |
| $I_{DD}$_IO | I/O Digital Current | 1080p (1920 × 1080), crop+bin2+scaling, 60 fps | – | 18 | 18 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 93 | 103 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 24 | 24 | |
| $I_{DD}$_IO | I/O Digital Current | 1080p LP (1920 × 1080), crop + bin2 + scaling, 60 fps | – | 18 | 18 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 70 | 77 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 24 | 24 | |
| $I_{DD}$_IO | I/O Digital Current | 720p (1280 × 720), crop + skip2 + scaling, 60 fps | – | 12 | 12 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 82 | 92 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 20 | 20 | |
| $I_{DD}$_IO | I/O Digital Current | 3M (2000 × 1500), crop + skip2 + scaling | – | 12 | 12 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 99 | 107 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 20 | 20 | |
| $I_{DD}$_IO | I/O Digital Current | 3M LP (2000 × 1500), crop + skip2 + scaling | – | 12 | 12 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 79 | 86 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 20 | 20 | |
| $I_{DD}$_IO | I/O Digital Current | VGA (640 × 480), crop + skip4 + scaling, 120 fps | – | 17 | 17 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 82 | 91 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 23 | 23 | |
| $I_{DD}$_IO | I/O Digital Current | 720p (1280 × 720), crop + skip2 + scaling, 120 fps | – | 28 | 28 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 116 | 128 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 25 | 25 | |

**Table 32. DC ELECTRICAL DEFINITIONS AND CHARACTERISTICS** (continued)
($f_{EXTCLK}$ = 25 MHz; $V_{AA}$ = 2.7 V; $V_{AA}$_PIX = 2.7 V; $V_{DD}$_IO = 1.8 V; $V_{DD}$ (Digital Core) = 1.2 V; $V_{DD}$_PHY = 1.2 V; Output load = 68.5 pF; $T_J$ = 60°C)

| Symbol | Parameter | Condition | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| **OPERATING CURRENT** | | | | | | |
| $I_{DD}$_IO | I/O Digital Current | QVGA (320 × 240), bin4skip4 240 fps | – | 17 | 17 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | 60 | 66 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | 23 | 23 | |
| **STANDBY CURRENT** | | | | | | |
| $I_{DD}$_IO | I/O Digital Current | XSHUTDOWN signal low (EXTCLK on) | – | – | 13 | μA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | – | 27 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | – | 10 | |
| $I_{DD}$_IO | I/O Digital Current | XSHUTDOWN signal low (EXTCLK off) | – | – | 13 | μA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | – | 27 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | – | 10 | |
| $I_{DD}$_IO | I/O Digital Current | STANDBY current when asserting R0x0100 = 0 (EXTCLK on) | – | – | 0.5 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | – | 14 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | – | 0.5 | |
| $I_{DD}$_IO | I/O Digital Current | STANDBY current when asserting R0x0100 = 0 (EXTCLK off) | – | – | 0.5 | mA |
| $I_{DD}$/$I_{DD}$_PHY/ $I_{DD}$_ANA | PHY/Digital Current | | – | – | 5 | |
| $I_{AA}$/$I_{AA}$_PIX | Analog/Pixel Current | | – | – | 0.5 | |

NOTE: This table specifies the typical measured currents with AR1335 sensor operating under typical light conditions. The per supply currents are measured with the typical and maximum supply voltages applied to the sensor.

### Absolute Maximum Ratings

**WARNING:** Stresses greater than those listed below may cause permanent damage to the device. Exposure to absolute maximum rating conditions for extended periods may affect reliability.
This is a stress rating only, and functional operation of the device at these or any other conditions above the nominal operating specification is not implied.

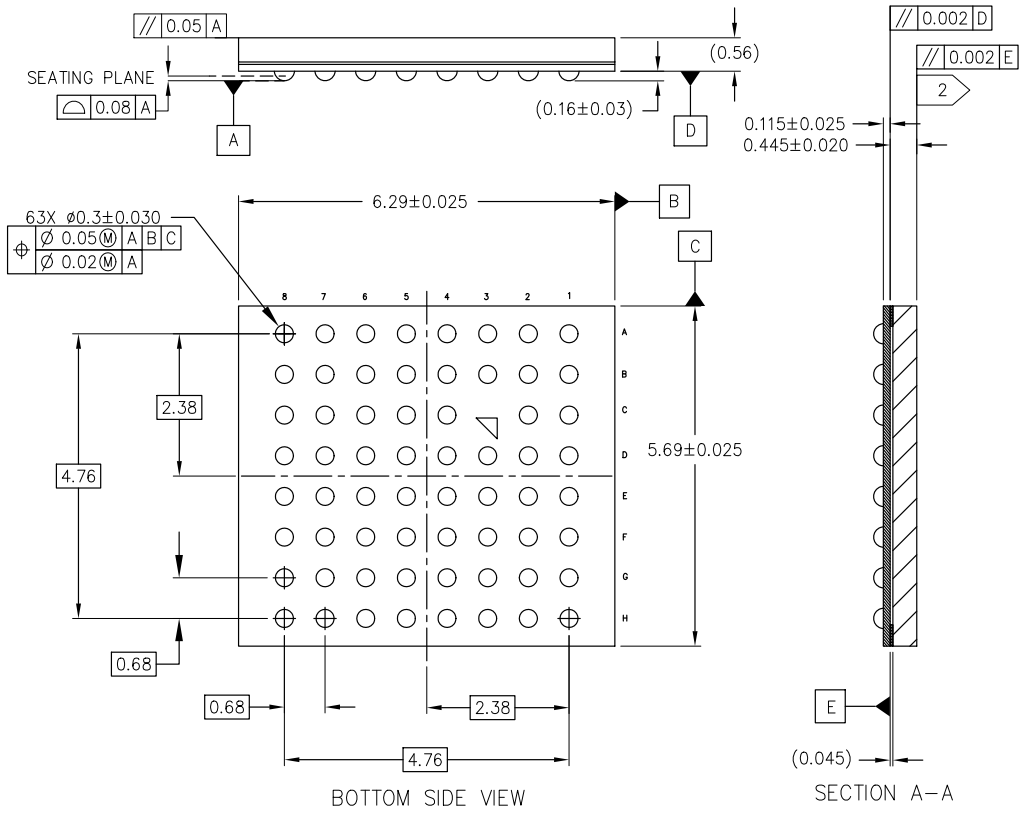
**Table 33. ABSOLUTE MAXIMUM RATINGS**

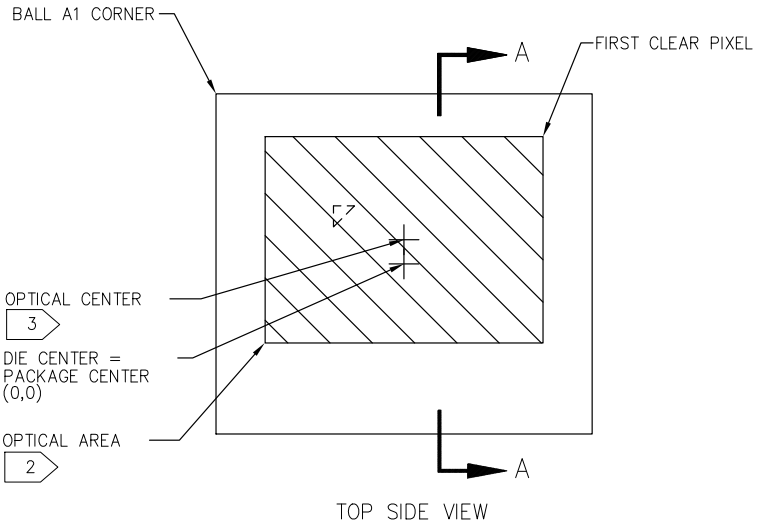
| Parameter | Min | Max | Unit |
|---|---|---|---|
| $V_{DD}$ & $V_{DD}$_PHY (1.2 V Domain) | –0.3 | 1.8 | V |
| $V_{DD}$_IO (1.8 V Domain) | –0.3 | 2.7 | V |
| $V_{AA}$ & $V_{AA}$_PIX (2.7 V Domain) | –0.3 | 3.5 | V |
| Operating Temperature (at Junction) | –30 | 70 | °C |
| Storage Temperature | –40 | 85 | °C |

Stresses exceeding those listed in the Maximum Ratings table may damage the device. If any of these limits are exceeded, device functionality should not be assumed, damage may occur and reliability may be affected.

**ODCSP63 6.29x5.69**
CASE 570BU
ISSUE O

DATE 19 OCT 2015




| NOTES | |
|---|---|
| 1 | DIMENSIONS IN MM. DIMENSIONS IN ( ) ARE FOR REFERENCE ONLY. |
| 2 | MAXIMUM ROTATION OF OPTICAL AREA RELATIVE TO PACKAGE EDGES: 0.1°.<br>MAXIMUM TILT OF OPTICAL AREA RELATIVE TO SUBSTRATE PLANE D: 2um.<br>MAXIMUM TILT OF COVER GLASS RELATIVE TO OPTICAL AREA PLANE E: 2um. |
| 3 | OPTICAL CENTER OFFSET FROM PACKAGE CENTER; X=0 Y=0.405 ±0.075 WITH RESPEC TO PACKAGE CENTER. |

| ISSUE | REVISION | DATE |
|---|---|---|
| O | RELEASED FOR PRODUCTION FROM APTINA POD# AR1335 TO ON SEMICONDUCTOR. REQ. BY D. TRUHITTE. | 19 OCT 2015 |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

**ON Semiconductor** and ON are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

ON Semiconductor and ON are trademarks of Semiconductor Components Industries, LLC dba ON Semiconductor or its subsidiaries in the United States and/or other countries. ON Semiconductor owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of ON Semiconductor's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent−Marking.pdf. ON Semiconductor reserves the right to make changes without further notice to any products herein. ON Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does ON Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using ON Semiconductor products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by ON Semiconductor. "Typical" parameters which may be provided in ON Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. ON Semiconductor does not convey any license under its patent rights nor the rights of others. ON Semiconductor products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use ON Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold ON Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that ON Semiconductor was negligent regarding the design or manufacture of the part. ON Semiconductor is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

## PUBLICATION ORDERING INFORMATION

**LITERATURE FULFILLMENT**:
Literature Distribution Center for ON Semiconductor
19521 E. 32nd Pkwy, Aurora, Colorado 80011 USA
**Phone**: 303−675−2175 or 800−344−3860 Toll Free USA/Canada
**Fax**: 303−675−2176 or 800−344−3867 Toll Free USA/Canada
**Email**: orderlit@onsemi.com

**N. American Technical Support**: 800−282−9855 Toll Free USA/Canada
**Europe, Middle East and Africa Technical Support:**
Phone: 421 33 790 2910

**ON Semiconductor Website**: **www.onsemi.com**

**Order Literature**: http://www.onsemi.com/orderlit

For additional information, please contact your local Sales Representative

◊